



Titre: L'analyse logique de données appliquée à la maintenance

Title: conditionnelle

Auteur: David Salamanca

Author:

Date: 2008

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Salamanca, D. (2008). L'analyse logique de données appliquée à la maintenance conditionnelle [Mémoire de maîtrise, École Polytechnique de Montréal].

Citation: PolyPublie. <https://publications.polymtl.ca/8363/>

 **Document en libre accès dans PolyPublie**

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/8363/>

PolyPublie URL:

**Directeurs de
recherche:**

Advisors:

Programme: Non spécifié

Program:

UNIVERSITÉ DE MONTRÉAL

L'ANALYSE LOGIQUE DE DONNÉES
APPLIQUÉE À LA MAINTENANCE CONDITIONNELLE

DAVID SALAMANCA
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE INDUSTRIEL)

AOÛT 2008



Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 978-0-494-46079-5

Our file Notre référence

ISBN: 978-0-494-46079-5

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

L'ANALYSE LOGIQUE DE DONNÉES
APPLIQUÉE À LA MAINTENANCE CONDITIONNELLE

présenté par: SALAMANCA David

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury constitué de:

Mme CHERIET Farida, ing., Ph.D., présidente

Mme YACOUT Soumaya, D.Sc., membre et directrice de recherche

M. OUALI Mohamed-Salah, Doctorat, membre

À mon épouse Suzette et à ma famille,
pour votre encouragement et votre patience.

REMERCIEMENTS

Je tiens à remercier madame Soumaya Yacout dont l'encadrement et le soutien m'ont motivé à poursuivre ce travail de recherche. Merci pour les nombreuses opportunités d'enrichissement professionnel que vous m'avez présentées lors de mes études.

Mes remerciements vont aussi aux professeurs du département de mathématiques et de génie industriel, grâce à qui j'ai acquis de précieuses connaissances lors de mon passage à l'École Polytechnique de Montréal.

Je remercie également tout le personnel du département, spécialement madame Suzanne Guindon pour ses informations précises et son soutien lors des démarches administratives.

Je ne saurai oublier Rafael Pires, un collègue, mais avant tout un ami que j'ai eu le privilège de rencontrer à l'école.

RÉSUMÉ

La maintenance conditionnelle est la planification des actions de maintenance selon l'état de la machine. Cependant, l'état n'est pas directement observable et doit être déterminé à partir des indicateurs prélevés de la machine à un instant donné (observation). L'Analyse Logique de Données (LAD, par son acronyme en anglais, *Logical Analysis of Data*) est une technique d'exploration de données qui permet de classifier les observations selon l'état associé.

Nous considérons que plusieurs avantages du LAD rendent très prometteuse son application à la maintenance conditionnelle : le LAD peut s'appliquer à une grande variété de problèmes de classification, les connaissances extraites sont compréhensibles pour les experts de la maintenance et de plus, le LAD performe bien par rapport au bruit et aux erreurs de mesure. Plus important encore, la précision de classification du LAD est comparable et souvent supérieure à celle d'autres méthodes de classification, (Hammer, P.L., 2000).

Le but de ce mémoire est l'exploration de l'utilisation du LAD à la maintenance conditionnelle ainsi que la conception d'un logiciel permettant d'automatiser l'analyse et la classification des observations de maintenance selon cette approche. À notre connaissance, l'application du LAD n'a jamais été étudiée dans ce contexte.

Ce mémoire est structuré de la façon suivante : dans la section introduction, les fondements théoriques de la maintenance conditionnelle et de l'analyse logique de données sont présentés. Dans le chapitre 2, les différentes étapes du LAD sont décrites et expliquées à l'aide d'exemples.

La conception des différents modules du logiciel ainsi que les détails d'implémentation tels que des diagrammes et des algorithmes sont présentés dans le chapitre 3. Dans le chapitre 4, deux applications de maintenance sont étudiées à l'aide du logiciel construit. Finalement, la section conclusions présente une vue sommaire des avantages et des défis de l'application de la méthode du LAD au domaine de la maintenance conditionnelle. Des recherches futures sont également suggérées en vue d'explorer certains aspects qui n'ont pas été traités dans ce travail, ainsi que des améliorations potentielles au logiciel développé en vue d'augmenter sa flexibilité et sa précision.

ABSTRACT

Condition-based maintenance consists of planning maintenance tasks depending on the machine state.

However this state is not directly observable and must be determined from indicators' measurements taken from the monitored machine at a given moment (observation). Logical Analysis of Data (LAD) is a data-mining technique that allows classification of observations according to the corresponding state.

We show that LAD's use to condition-based maintenance can be fruitful as it exhibits several advantages: LAD can be used on a great variety of classification problems, extracted knowledge is comprehensible by maintenance experts and LAD reacts well to signal noise and measurement errors. Most important, is the fact that LAD's classification accuracy is comparable and often superior to other classification methods.(Hammer, P.L. et al., 2000)

The main objective of this thesis is the exploratory use of LAD to condition-based maintenance as well as development of a software application that allows automation of the analysis and classification of maintenance observations according to the aforementioned technique. To the best of our knowledge, LAD has never been used in the context of condition-based maintenance.

This research paper is organized as follows: In the introduction, the theoretical foundation of condition-based maintenance and of LAD is presented. In chapter 2, LAD's different stages are described and illustrated with examples.

The design and implementation of the different modules of which the software application is comprised, accompanied by diagrams and the corresponding algorithms are presented in chapter 3.

In the following chapter, two maintenance applications are studied using the software application that was developed. Finally under "Conclusions" we present a summarized view of the advantages as well as the challenges and limitations faced when building and using the software in the condition-based maintenance context. We also include a series of future research proposals which cover aspects that are not explored under the present research effort, as well as suggest ways to improve the performance and flexibility of the software that was built.

TABLE DE MATIÈRES

DÉDICACE	iv
REMERCIEMENTS	v
RÉSUMÉ	vi
ABSTRACT	viii
LISTE DES TABLEAUX	xii
LISTE DES FIGURES	xv
LISTE DES ÉQUATIONS	xvii
LISTE DES ANNEXES	xviii
INTRODUCTION	1
CHAPITRE 1 LA MAINTENANCE CONDITIONNELLE	6
CHAPITRE 2 L'ANALYSE LOGIQUE DE DONNÉES	18
2.1 Méthodologie du LAD	22
2.1.1 La conversion binaire de données	27
2.1.2 La génération de patrons	39
2.1.3 La fonction discriminante	52
2.1.4 La classification	57

CHAPITRE 3	CONCEPTION DU LOGICIEL.....	65
3.1	La conversion binaire de données.	71
3.2	Implémentation de l'algorithme de génération de patrons.....	75
3.3	Formation de la fonction discriminante et classification.....	82
CHAPITRE 4	APPLICATIONS À LA MAINTENANCE CONDITIONNELLE....	85
4.1	Maintenance conditionnelle d'un mécanisme de roulement	86
4.2	Maintenance conditionnelle d'un transformateur de courant.....	100
CONCLUSIONS ET RECHERCHES FUTURES		113
RÉFÉRENCES.....		121
ANNEXES.....		127

LISTE DES TABLEAUX

Tableau 2.1	Observations de l'étude de prescription de verres de contact (Cendrowska, J., 1987).....	25
Tableau 2.2	Observations de l'étude de prescription de verres de contact à deux états, basé sur (Cendrowska, J., 1987).....	26
Tableau 2.3	Observations d'un roulement mécanique. (Sun Q. et al, 2004).....	31
Tableau 2.4	Roulement mécanique: Indicateurs ordonnancés et états associés.....	31
Tableau 2.5	Roulement mécanique: Données en format binaire	39
Tableau 2.6	Synthèse du nombre de termes vs degré maximal de patrons générés.....	43
Tableau 2.7	Étude de prescription de verres de contact - Génération de patrons	49
Tableau 2.8	Les patrons générés et leur couverture - Étude de prescription de verres de contact.....	50
Tableau 2.9	Poids normalisés - Étude de prescription de verres de contact	56
Tableau 2.10	Ensemble de tests - Étude de prescription de verres de contact.....	61
Tableau 2.11	Résultats de classification - Étude de prescription de verres de contact..	62
Tableau 4.1	Observations de maintenance conditionnelle d'un mécanisme de roulement (Sun Q., et al., 2004).....	87
Tableau 4.2	Plan d'expériences pour l'apprentissage et le test du mécanisme de roulement	88

Tableau 4.3	Observations en format binaire pour le premier essai - Mécanisme de roulement	89
Tableau 4.4	Synthèse de la conversion binaire des observations du mécanisme de roulement	89
Tableau 4.5	Cinq patrons positifs et cinq premiers patrons négatifs du premier essai – Mécanisme de roulement.....	91
Tableau 4.6	Synthèse de la génération de patrons - Mécanisme de roulement.....	92
Tableau 4.7	Résultats numériques de classification - Mécanisme de roulement.....	94
Tableau 4.8	Exemple de calcul de la zone d'indécision et des zones de décision - Mécanisme de roulement.....	98
Tableau 4.9	Résultats finaux de la classification - Mécanisme de roulement	98
Tableau 4.10	Observations de l'historique - Transformateur de courant.....	101
Tableau 4.11	Synthèse de l'étape d'apprentissage - Transformateur de courant	104
Tableau 4.12	Observations de tests - Transformateur de courant.....	105
Tableau 4.13	Résultats de classification du 1 ^{er} essai : État de décharge de haute énergie et normal - Transformateur de courant.....	106
Tableau 4.14	Résultats de classification du 2 ^{ème} essai : États de décharge de basse énergie et normal - Transformateur de courant	106
Tableau 4.15	Résultats de classification du 3 ^{ème} essai : États de réchauffement thermique - et normal - Transformateur de courant.....	107
Tableau 4.16	Résultats de classification - Transformateur de courant	108
Tableau 4.17	Synthèse de performance de classification - Transformateur de courant.....	111

Tableau A.1	Observations en format binaire - Étude de verres de contact.....	127
Tableau B.1	Patrons générés au cours du premier essai - Mécanisme de roulement...	128
Tableau C.1	Données d'apprentissage. État de décharge de haute énergie et état normal - Transformateur de courant	133
Tableau C.2	Données de test. État de décharge de haute énergie et état normal - Transformateur de courant	134
Tableau C.3	Données d'apprentissage - État de décharge de basse énergie et état normal - Transformateur de courant.....	135
Tableau C.4	Données de test - État de décharge de basse énergie et état normal - Transformateur de courant.....	136
Tableau C.5	Données d'apprentissage - État de réchauffement thermique et état normal - Transformateur de courant.....	137
Tableau C.6	Données de test. État de réchauffement thermique et état normal - Transformateur de courant	139

LISTE DES FIGURES

Figure 1.1	Processus de défaillance à deux étapes. Source: Adapté au français de (Wang, 2007).....	7
Figure 2.1	Zones de Classification des observations	58
Figure 2.2	Approche graphique de classification par comparaison avec l'ensemble d'apprentissage.....	60
Figure 3.1	Modules du logiciel LAD-CBM.....	66
Figure 3.2	Diagramme d'objets global.....	68
Figure 3.3	Diagramme d'objets de l'algorithme LAD et de classes de support	69
Figure 3.4	Diagramme de séquence des interactions entre la classe interface et la classe binarisation.....	72
Figure 3.5	Diagramme de séquence de la binarisation étape par étape	73
Figure 3.6	Diagramme de séquence sur l'analyse de chaque point de coupure.....	74
Figure 3.7	Génération de patrons 1 ^{ère} partie.....	78
Figure 3.8	Génération de patrons, 2 ^{ème} partie	79
Figure 3.9	findIfInCandidate, référence de support à la génération de patrons.....	80
Figure 3.10	AnalyzeIFCandidateOrPrime, référence de support à la génération de patrons	81
Figure 3.11	Formation de la fonction discriminante et classification de nouvelles observations.....	84

Figure 4.1 Diagramme d'un roulement mécanique à billes.....	86
---	----

LISTE DES ÉQUATIONS

Équation 2.1	Point de coupure	32
Équation 2.2	Nombre de combinaisons différentes de d éléments parmi n	43
Équation 2.3	Nombre de termes générés jusqu'au degré d	43
Équation 2.4	Fonction discriminante	52
Équation 2.5	Poids de patrons	53
Équation 2.6	Somme de poids normalisés des patrons positifs et des patrons négatifs	54
Équation 2.7	Poids normalisés de patrons positifs	54
Équation 2.8	Poids normalisés de patrons négatifs	54
Équation 2.9	Fonction discriminante normalisée	55
Équation 2.10	Les poids normalisés de patrons - Étude de prescription de verres de contact	56

LISTE DES ANNEXES

ANNEXE A : ÉTUDE DE PRESCRIPTION DE VERRES DE CONTACT	127
ANNEXE B : APPLICATION DE CBM AU MÉCANISME DE ROULEMENT	128
ANNEXE C : APPLICATION DE CBM AU TRANSFORMATEUR DE COURANT....	
.....	133

INTRODUCTION

La fiabilité des machines industrielles est un sujet autour duquel s'effectue beaucoup de recherches, tant dans les modèles théoriques généraux que dans les études et la modélisation pour des systèmes de production spécifiques.

L'étude de fiabilité industrielle a pour but de rallonger la vie utile des machines industrielles et d'assurer leur disponibilité dans la chaîne de production, tout en minimisant des arrêts non planifiés et le risque de défaillances critiques. Les programmes de maintenance sont des plans d'action qui visent à conserver un équipement ou un système dans un état fonctionnel. La maintenance inclut également les actions correctives pour remettre un équipement en fonction et gérer son cycle de vie (Dekker R., 1996).

Les premières pratiques de maintenance datent de 1950 (Pintelon, L.M. et al. 1992). À cette période, la maintenance préventive basée sur des intervalles fixes au bout desquels on réalisait des rechanges de pièces où des réparations, était pratique courante chez les entreprises pour réduire les temps d'arrêt non-planifiés.

Malgré le fait qu'il existe des modèles avancés pour mettre en place des plans pour prévenir les pannes, il y a encore dans l'industrie deux courants extrêmes selon le type d'approche préconisée : la maintenance après panne ou corrective (approche passive) et la maintenance préventive basée sur l'âge de la machine (approche proactive).

La maintenance d'après-panne est coûteuse non seulement en termes de productivité en raison des arrêts non planifiés, mais elle exige également la présence continue d'une équipe de maintenance sur les lieux en vue d'éviter des arrêts prolongés. Comme les actions de maintenance ne sont pas planifiées à l'avance, le temps d'arrêt est plus long vu que les pièces de rechange ne sont pas toujours disponibles sur place. Avec cette politique de maintenance, une panne peut survenir en tout temps et retarder le carnet de production. La production juste-à-temps est très risquée dans ce cas-ci.

La maintenance planifiée selon l'âge de l'équipement ou à intervalles réguliers est efficace dans la mesure où les pannes sont évitées si l'intervalle de temps entre deux interventions de maintenance est assez petit, mais les arrêts planifiés seront, peut-être, très fréquents et des pièces qui pourraient encore avoir une vie utile sont souvent remplacées précocement. Il existe également d'autres variantes de cette méthode, permettant d'établir un plan de maintenance qui se base sur l'historique de défaillances et qui tient compte de l'âge de la machine.

L'importance de la maintenance est la conséquence directe de l'automatisation accrue des milieux de production, ainsi que les effets négatifs et parfois critiques que les pannes peuvent avoir sur une chaîne de production. (Wiseman, 1990) a conclu lors de ses recherches, que les dépenses liées à la maintenance de machines dans des entreprises aux États-Unis ont augmenté de 10 à 15% par année depuis 1979.

Les coûts élevés de la maintenance incitent l'adoption, le développement et l'amélioration continue de méthodes de gestion du cycle de vie des équipements. Le

coût de maintenance des équipements de certaines entreprises manufacturières peut être si important qu'il atteigne un montant similaire à leurs revenus (McKeon, 1998).

En 1970, les techniques et l'utilisation de la prise de mesures de certains indicateurs qui caractérisent l'état actuel des machines (la vibration ou la concentration de particules dans l'huile lubrifiante) ont été mises au point.

Les praticiens se sont aperçus que cette méthode était plus performante que la maintenance d'après panne ou sur la maintenance qui se basait uniquement sur l'âge de l'équipement pour faire les prédictions de la vie utile de l'équipement. Ses plus grands avantages sont :

1. Une intervention de maintenance n'est réalisée que lorsque l'état de la machine le requiert et en conséquence le nombre d'arrêts planifiés est diminué.
2. La surveillance continue de la machine permet d'intervenir avant qu'une panne non-planifiée ne se produise et permet souvent de planifier l'intervention de maintenance de façon à réduire les retards sur le carnet de production.
3. Le coût des pièces de rechange et des interventions de maintenance diminue, car les actions de maintenance sont pilotées par un besoin réel. (Martin, K.F., 1994).

(Al-Najas, 2003) a fait une étude sur l'amélioration de la productivité d'une compagnie par la mise en fonction de la maintenance conditionnelle basée sur des mesures de vibration. Cette étude qui s'est étalée sur trois ans, montre les avantages économiques qui découlent de l'implémentation d'un programme de maintenance conditionnelle dans une étude de cas dans l'industrie de pâtes à papier en Suède.

La maintenance conditionnelle génère un grand volume de données qui doit être analysé afin de planifier des actions de maintenance. L'analyse logique de données (LAD par ses sigles en anglais, *Logical Analysis of Data*) permet d'extraire de l'information structurelle à partir d'un historique de données afin de qualifier l'état de la machine lorsque de nouvelles données sont disponibles. Ainsi lorsque de capteurs sont utilisés pour faire le suivi d'une machine, le LAD analyse les nouvelles données disponibles et permet de qualifier l'état associé de la machine afin de planifier des actions de maintenance lorsque c'est nécessaire.

Le LAD est une technique d'exploration de données (« data mining » en anglais) proposée par P.L. Hammer qui présente des performances de classification comparables ou supérieures aux méthodes traditionnelles de classification tels que des réseaux neuronaux et des arbres de décision entre autres (P.L Hammer et al, 2000). Le LAD fournit non seulement un résultat de classification conforme aux données de l'historique, mais extrait également des connaissances sous forme de patrons logiques compréhensibles aux non-experts, ce qui permet de comprendre les phénomènes sous-jacents au problème traité. Cet avantage est d'une importance notable dans le domaine

de la maintenance conditionnelle, car comprendre les causes d'une défaillance est aussi importante que d'identifier le moment où elle se produit.

À notre connaissance, c'est la première fois que le LAD est utilisé dans le domaine de la maintenance conditionnelle. L'objectif de ce travail est d'étudier l'application du LAD à la maintenance conditionnelle, en vue de faire un pronostic sur la condition de la machine étudiée. À ce propos et en vue de construire un système de décisions automatisé, un logiciel permettant d'appliquer le LAD aux données de maintenance a été conçu.

Ce document se divise en quatre chapitres et une section de conclusions. Le chapitre I présente une synthèse des principaux concepts et des avancements récents dans le domaine de la maintenance conditionnelle. Le chapitre II explorera les origines historiques de la méthode de l'analyse logique de données et ses avantages. Les différentes étapes du LAD sont également décrites et expliquées à l'aide d'exemples. Le chapitre III décrit la méthodologie suivie lors de la conception et le développement du logiciel LAD-CBM et les étapes que comporte le LAD de façon plus détaillée. Le chapitre IV aborde l'utilisation du logiciel développé pour l'analyse de données et la classification de deux cas d'études de maintenance conditionnelle, soit des mesures de vibration d'un mécanisme de roulement et des mesures de gaz dissout dans un transformateur de courant. Finalement, dans la section de conclusions, nous présentons une synthèse du travail et des résultats, les défis soulevés et nous proposons des éléments de recherches futures.

CHAPITRE 1

LA MAINTENANCE CONDITIONNELLE

La maintenance conditionnelle fournit un cadre de décision afin de mettre en place une politique de maintenance qui dépend des conditions intrinsèques de la machine à un moment donné. Pour connaître l'état de la machine, on utilise des mesures des indicateurs telles que les fréquences de vibration, la température des composants, des données acoustiques et la pression et la concentration des métaux dans les lubrifiants entre autres.

Selon le but du plan de maintenance conditionnelle, il existe deux étapes différentes, le diagnostic et le pronostic. Dans le diagnostic, le but est de détecter, et dans certains cas, d'isoler la cause de la défaillance après que celle-ci soit survenue. Le pronostic, par contre, vise à prédire la probabilité ou le moment où une panne surviendra dans le futur et ainsi détecter une condition anormale avant qu'une panne ne se produise.

Les deux analyses sont complémentaires, et ce qui est souvent le cas, les résultats des analyses de diagnostic permettent d'améliorer la précision des analyses de pronostic.

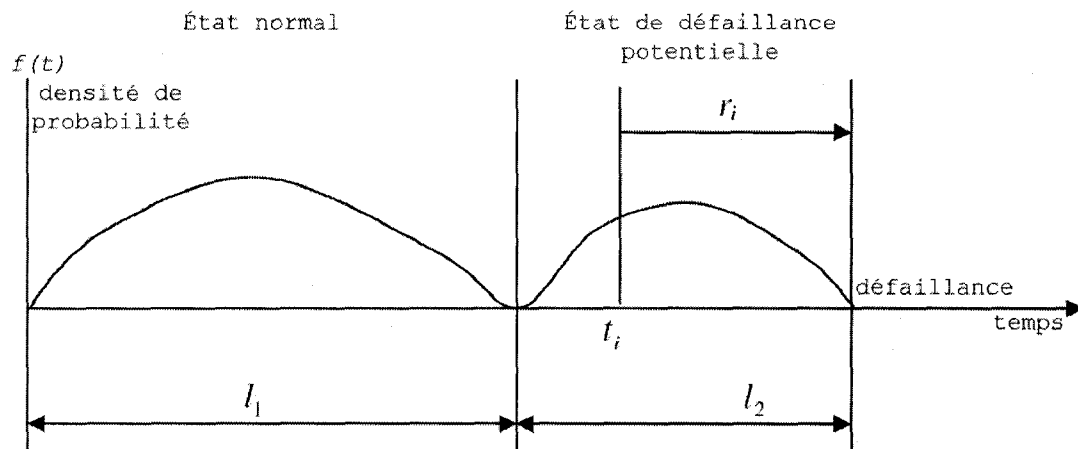


Figure 1.1 Processus de défaillance à deux étapes. Source: Adapté au français de (Wang, 2007)

Le modèle de pronostic à deux étapes (Christer, A.H., 1976) établit qu'une machine connaît dans son cycle de vie deux étapes. Dans l'étape de fonctionnement normal, la machine est exploitable et la probabilité de défaillance est faible. L'étape de défaillance potentielle commence à partir du moment où une défaillance est plus susceptible de se développer.

La Figure 1.1 illustre le processus de défaillance à deux étapes, où l'on a l_1 qui est la durée de l'étape de fonctionnement normal et l_2 qui est la durée de l'étape de défaillance à partir du moment de la transition jusqu'à ce que la défaillance survienne. Le temps t_i est le temps de l' $i^{\text{ème}}$ inspection à partir de l'état neuf, et r_i représente la durée résiduelle, ou le temps de vie utile qui reste à l'équipement jusqu'à ce que la défaillance se présente. En maintenance conditionnelle, on s'intéresse à connaître le moment de transition entre l'étape normale et l'étape de défaillance potentielle ainsi qu'à la vie

résiduelle de l'équipement au moment d'une inspection. Lors de cette inspection, la prise de mesures des indicateurs d'état permet d'évaluer la durée de vie résiduelle. (Wang, W., 2007)

Tout programme de maintenance conditionnelle est constitué de trois étapes fondamentales:

(Lee, J. et al., 2004)

- Acquisition de données : cueillette des mesures sur les indicateurs de l'état du système;
- Traitement de données : nettoyage et filtrage de données ainsi que l'extraction des caractéristiques, patrons et tendances à partir de ces données;
- Prise de décisions de maintenance : en tenant compte d'un ou plusieurs critères, tels que le coût total de possession ou la disponibilité, une décision de maintenance doit être prise à partir de connaissances extraites à l'étape précédente.

L'acquisition de données consiste à prélever des mesures ou des informations sur l'équipement à surveiller, selon une certaine fréquence d'échantillonnage. Dans ce contexte, il existe deux types de données : les données liées aux événements et les

données de surveillance de l'état. Les données liées aux événements correspondent à toutes les interventions (réparation, arrêt, démarrage) et aux circonstances qui ont engendré ces actions, comme une maintenance préventive, un état d'urgence, etc. Les données de surveillance d'état sont toutes les mesures des indicateurs de l'équipement ou des indicateurs de l'état.

La prise de données liées aux événements se fait généralement de façon manuelle. Lorsqu'une situation spéciale se produit, l'opérateur inscrit sur le registre la date, la gravité de l'événement (le mode de défaillance), la description de ce qui est arrivé (les effets de la défaillance) et les actions qui ont été prises.

D'autre part, les mesures de surveillance d'état sont prises par des capteurs de façon automatique et sont généralement intégrées au système de gestion de maintenance de l'entreprise. Il existe une grande variété de capteurs tels que senseurs optiques, acoustiques, et ultrasoniques entre autres adaptés aux différents types d'indicateurs à surveiller. Le type de données recueillies et analysées peuvent correspondre tant au domaine fréquentiel qu'au temporel.

La recherche effectuée autour du sujet de prise de mesures est souvent spécifique au type de mécanisme ou de l'équipement étudié. Dans l'article de (Tandon, N., 1999), on trouve une revue sur les principales méthodes de mesure pour la détection de défaillances dans les billes de roulement qui sont un composant primordial dans les mécanismes rotationnels. R.J Kuoppalla fait une synthèse des indicateurs les plus

fréquemment utilisés en maintenance conditionnelle des mécanismes de rotation, ainsi que leur champ d'application (type de défaut surveillé) dans (K. Holmberg, 1991).

La technologie de capteurs a beaucoup évolué dernièrement (Jardine, A.K.S., 2006. Kirianaki, N.V., 2002). Les modèles récents sont relativement peu dispendieux et sont spécifiquement adaptés au type de données à cueillir. De ce fait, plusieurs senseurs peuvent récolter des données d'une même machine. Par exemple, dans le domaine de l'automobile, des senseurs pourraient être installés pour évaluer les billes de roulement des quatre roues afin d'étudier l'usure qui donne lieu aux défaillances.

Un cadre de travail a été proposé pour intégrer l'information de différentes sources grâce aux techniques d'exploration de données pour découvrir de patrons et de relations entre les indicateurs (Raheja, D., 2006).

Dans plusieurs applications, la fusion de données de multiples capteurs donne lieu à des meilleurs résultats de diagnostic que dans le cas où seulement un capteur était utilisé. (Liu, Q., 2001)

Une fois que l'on possède les mesures des observations, on doit procéder au nettoyage de données. Les sources d'erreur les plus communes sont l'erreur humaine et le mal fonctionnement des capteurs. Généralement, un contrôle manuel à l'aide de graphiques où les points aberrants ressortent, permet de détecter ces comportements normaux. Les observations qui contiennent des mesures erronées doivent être négligées de toute analyse subséquente.

Une fois que les mesures qui contiennent des erreurs ont été détectées et éliminées, on procède à l'analyse de données selon leur type.

Il existe trois types de données : type valeur, type de signal d'onde et type multidimensionnel. La mesure de type valeur, est une valeur ponctuelle unique prise à un moment donné telle que la température ou l'humidité. La donnée de type d'onde consiste en une série de mesures prises dans le temps telle qu'une donnée acoustique. Les données multidimensionnelles sont souvent des images telles que des thermographes infrarouges entre autres.

Le traitement de signaux permet justement d'extraire les caractéristiques de données de type d'onde et de type multidimensionnel. Ces caractéristiques sont, entre autres, des valeurs de crête et des données statistiques sur les données de base. (A.K.S. Jardine, et al. 2006).

Un grand nombre de méthodes ont été mises au point pour faire le traitement de données afin d'extraire de caractéristiques utiles. Parmi ces techniques, on trouve des techniques de traitement de signal (domaine temporel, fréquentiel ou mixte) ainsi que de méthodes basées sur la reconnaissance de formes et de traitement d'images.

Des approches statistiques, telles que de l'analyse multivariable dont l'analyse de composants principaux et l'analyse de composants indépendants, sont aussi utilisées pour extraire des informations utiles à partir d'un nombre considérable d'indicateurs. En (Stellman, C.M, 1999), les auteurs ont utilisé l'analyse multi-variable pour étudier la

dégradation de lubrifiants, alors qu'une autre étude (Allgood, G.O., 2000) propose un système de diagnostic basé sur l'analyse de composants principaux (PCA).

Une fois que l'analyse sur les caractéristiques utiles des mesures des indicateurs a été effectuée, le module de support aux décisions de maintenance permet aux praticiens de planifier des actions préventives de maintenance au moment approprié.

Trois types de méthodes sont appliqués pour effectuer la prise de décisions de maintenance: les méthodes statistiques, les méthodes basées sur l'intelligence artificielle et les méthodes basées sur des modèles mathématiques (Simani, S., 2003).

L'approche de modélisation mathématique se base sur des abstractions mathématiques et physiques de la machine. Parmi les méthodes de modélisation mathématique utilisées en maintenance conditionnelle, on trouve des méthodes de génération de résidus et des techniques d'identification des paramètres du système (Gertler, J.J., 1998). En particulier, les filtres de Kalman permettent d'extraire des résidus qui sont des indicateurs de défaillances dans l'équipement. (Sekhar, A.S., 2004)

L'approche de modélisation mathématique, possède des performances qui sont dans certains cas supérieures aux autres approches (Jardine, 2006). La grande limitation de l'approche basée sur des modèles, est qu'il est difficile, voire impossible de déduire des modèles mathématiques pour des systèmes complexes.

Les méthodes statistiques s'appuient sur le Contrôle de Processus Statistiques (CPS), qui permet de surveiller un processus à l'aide des mesures statistiques et de graphiques afin

de décerner des patrons de fonctionnement. Le CPS utilise de cartes de contrôle, des histogrammes de fréquence et des diagrammes de cause à effet pour permettre d'identifier de façon graphique tout comportement qui s'éloigne de la normale. (Fugate, M.L. et al., 2001)

L'analyse de composants principaux (ACP) est une méthode statistique multi-variée qui permet de constituer de groupes d'observations qui correspondent aux différents types d'état dans lesquels peut se trouver une machine. Généralement, les groupes sont déterminés en fonction des distances entre les points. La méthode vise à augmenter la distance entre deux points appartenant à deux états différents de la machine, et d'en rapprocher les observations qui appartiennent à un même état de la machine. (Stellman et al. 1993, Allgood et al, 1994)

D'autres approches statistiques utilisées en maintenance conditionnelle sont les machines de vecteurs de support et les chaînes de Markov cachées. (Li, Z. et al., 2005. Hong, W.C. et al., 2006)

Des méthodes d'intelligence artificielle qui se basent sur l'extraction des connaissances automatisées sont de plus en plus utilisées et présentent de gains de performance par rapport aux méthodes conventionnelles (Jardine, 2006). (Peng, Y., 2004) a utilisé des arbres de décision basés sur la logique floue pour améliorer la performance des processus inductifs d'apprentissage pour la maintenance conditionnelle. En (Amjady, N., 1999), les auteurs proposent l'utilisation de réseaux de neurones pour l'évaluation de la fiabilité des transformateurs et des générateurs électriques et citent, parmi les avantages,

la rapidité de calculs qui rend la technique très flexible aux conditions changeantes des machines.

Les réseaux de neurones requièrent des observations d'apprentissage pour trouver la fonction inconnue qu'elles veulent approximer, dans cas-ci la probabilité de défaillance ou la vie résiduelle de la machine étudiée. Un réseau de neurones consiste en un groupement de noyaux connectés d'une façon spécifique qui possède plusieurs entrées et sorties et qui est capable d'approximer une fonction non-linéaire.

Le processus d'ajustement de poids de neurones afin de converger vers cette fonction est connu comme processus d'apprentissage. La plupart des réseaux neuronaux tels que des réseaux neuronaux en cascade (CCNN, en anglais) ou des réseaux à boucle d'anticipation (Feedforward neural network) utilisent l'apprentissage supervisé, c'est-à-dire que lors de la phase d'apprentissage, en plus d'avoir les mesures des indicateurs (entrées), l'état correspondant de la machine (sorties) doit être également connu.

Les limitations des réseaux neuronaux sont le fait que le réseau obtenu après l'entraînement est, à toute fin utile, une boîte noire qui donne peu ou pas d'indication sur le phénomène de défaillance en question et se limite à donner la sortie associée à une entrée donnée, et le fait que le processus d'apprentissage est de lente convergence requiert la connaissance à priori des sorties (Siddique, A., 2003).

Des études comparatives de performance ont été effectuées pour comparer la performance des réseaux de neurones avec d'autres techniques d'intelligence artificielle telles que des machines de support des vecteurs avec des algorithmes génétiques dans la

classification des défaillances des billes de roulement (Samanta, B., 2003). (Spoerre, J.K., 1997) a utilisé des CCNN pour la classification des défaillances de mécanismes rotationnels.

D'autre part, les systèmes experts (ES, sigle en anglais) utilisent des connaissances d'experts pour effectuer un raisonnement grâce à un moteur d'inférence. Trois types d'ES sont utilisés en CBM : ceux qui effectuent un raisonnement basé sur des règles, selon un raisonnement basé sur des cas et ceux qui font leur raisonnement basé sur des modèles.

L'ES connaissent le désavantage de l'explosion combinatoire. Celle-ci consiste en ce que plus le nombre d'indicateurs surveillés est grand, plus le nombre de règles et la complexité de calcul augmentent, et ce, de façon exponentielle. Dans le contexte de maintenance conditionnelle, on trouve des études qui utilisent les systèmes experts comme méthodes de prise de décision : (Yoon, Y.H., 1993) en raisonnement basé sur des règles, (Wen, Z.Y., 2003) en raisonnement par des cas et (Baig, M.F., 1998) en raisonnement par des modèles.

Certaines études de maintenance utilisent des méthodologies qui combinent plusieurs des techniques énoncées en vue d'augmenter la précision de résultats ou de contourner les limitations d'une méthode particulière d'intelligence artificielle. (Sohn, H., 2002) a développé une méthode hybride de classification de dommages sous conditions environnementales changeantes.

Cette méthode utilise un modèle mathématique autorégressif pour l'extraction de caractéristiques utiles à partir de données brutes; ensuite un réseau neuronal est utilisé pour normaliser des données et éliminer l'influence de l'environnement sur les mesures. Finalement, une technique statistique basée sur des tests d'hypothèses est utilisée pour l'inférence sur l'état de la machine.

L'exploration de données (data mining en anglais) est considéré être un domaine qui combine la statistique, l'intelligence artificielle et la recherche sur des bases des données. (Pregibon, D, 1997). Les techniques d'exploration de données font l'extraction des modèles concis et des relations à partir d'un grand nombre de données, en utilisant des algorithmes d'apprentissage entièrement automatisés.

Les techniques d'exploration de données peuvent être catégorisées selon le type de base de données utilisée (modèle de base de données relationnelle ou orientée objets), la nature de la connaissance à extraire (données relationnelles ou basé sur modèle transactionnel) et les techniques employées pour effectuer l'extraction de ces connaissances (statistique, intelligence artificielle) (Lee, S.J. et al., 2001).

La littérature qui décrit l'application des techniques d'exploration de données à la maintenance conditionnelle est récente et la recherche très spécifique au domaine d'application (Raheja, D., 2006). (Wu, 2004) propose l'utilisation de techniques d'exploration de données dans un système de maintenance pour un transformateur électrique. En (Raheja, D., 2006), les auteurs proposent une architecture hybride

d'exploration et fusion de données provenant de différents capteurs rattachés à la machine.

Le but de ce mémoire est d'explorer l'utilisation d'une méthode d'exploration de données basée sur la logique booléenne et l'analyse combinatoire, dans la maintenance conditionnelle. L'analyse logique de données est une technique qui permet l'extraction des caractéristiques structurelles des données historiques afin d'utiliser ces connaissances dans la classification de l'état actuel de la machine en question.

Le chapitre II de ce mémoire explorera les origines historiques de la méthode et ses avantages. Les différentes étapes du LAD seront également décrites et expliquées à l'aide d'exemples.

CHAPITRE 2

L'ANALYSE LOGIQUE DE DONNÉES

Le LAD est une technique d'exploration de données qui permet d'extraire des connaissances à partir de données brutes. Les connaissances sont extraites sous forme de patrons logiques. Quand les patrons ont été extraits, la classification des nouvelles données se réalise avec peu ou pas d'intervention humaine.

Cette caractéristique que le LAD partage avec d'autres techniques d'exploration de données, ainsi que la capacité de traiter des volumes considérables des données ont suscité l'intérêt des chercheurs récemment. En termes généraux, les techniques pour effectuer le diagnostic de défaillances des machines basées sur la reconnaissance de patrons sont constituées des étapes suivantes (Sun, Q., 2004):

- Acquisition de données;
- Identification des indicateurs;
- Fusion des indicateurs;
- Apprentissage non-linéaire;
- Classification;
- Prise de décision de diagnostic.

Les données mesurées lors de l'acquisition des données (abordée au chapitre précédent), sont utilisées pour faire l'identification des indicateurs. Cette étape vise à identifier les indicateurs d'importance parmi la grande quantité de signaux des capteurs. Généralement, ces indicateurs peuvent être des paramètres statistiques extraits des données dans le domaine temporel ou fréquentiel.

De plus, en présence de données qui proviennent de plusieurs capteurs, des techniques de fusion des indicateurs permettent de réduire le nombre de paramètres à analyser, tandis que l'apprentissage non linéaire est utilisé dans le but d'identifier les tendances et de découvrir les relations entre les données.

Il est important de noter que l'apprentissage d'un algorithme de reconnaissance de patrons est supervisé car en plus de fournir des mesures des indicateurs, l'état associé de la machine doit également être connu à priori. Ceci veut dire que l'on doit avoir un historique des mesures et l'état associé avant que l'apprentissage et la classification puissent être effectués.

Les résultats de la phase d'apprentissage sont un ensemble de patrons qui caractérise les observations de l'historique, et en conséquence ont un pouvoir de classification sur des nouvelles observations pour lesquelles l'état n'est pas connu.

L'étape de classification consiste à fournir un nouvel ensemble de mesures des indicateurs pour que l'algorithme déduise l'état actuel de la machine en se basant sur les patrons découverts à l'étape d'apprentissage.

Le LAD a été proposé pour la première fois par Peter L. Hammer de Rutgers University Center for Operations Research (RUTCOR) (Hammer, P.L., 1986). Le LAD comme l'analyse statistique, la reconnaissance de patrons, l'apprentissage des machines et les réseaux de neurones parmi d'autres, permet de résoudre les problèmes de classification des observations et d'extraction de patrons ou de tendances d'un grand volume de données.

L'avantage du LAD par rapport à d'autres méthodes d'intelligence artificielle réside dans son pouvoir explicatif. À l'aide de patrons logiques extraits de données historiques, la méthode fournit non seulement un résultat de classification mais il est aussi possible de comprendre comment le résultat a été produit. Le résultat de classification correspondant à une observation particulière peut être compris en faisant l'analyse de patrons validés par les mesures des indicateurs de cette observation.

Le LAD a des performances comparables aux méthodes établies qui sont utilisées actuellement en classification (Hammer, P.L., 2000). À notre connaissance, le LAD n'a jamais été utilisé dans le domaine de la maintenance conditionnelle. Il serait intéressant d'explorer l'utilité et les gains par rapport aux techniques utilisées actuellement et de comprendre la nature des défaillances à la lumière de patrons validés par la méthode LAD.

Le LAD dans son étape d'apprentissage, requiert les mesures des indicateurs c'est-à-dire les observations (une observation est une combinaison de mesures des indicateurs prise à un moment spécifique) et aussi l'état associé. Cet état qui décrit la condition de la

machine à un moment donné peut correspondre à une des deux valeurs suivantes: l'état de fonctionnement normal et l'état de défaillance potentielle.

Bien que les données d'état soient généralement négligées dans le contexte de la maintenance conditionnelle (le processus manuel de tenue du registre requiert une assiduité de la part de l'opérateur), ces données sont d'une grande utilité dans le contexte du LAD. Une grande partie des machines sont des systèmes partiellement observables, c'est-à-dire que l'on peut mesurer la valeur des indicateurs de la machine, mais l'état n'est pas directement observable ou mesurable. Dans ce cas-ci, l'état doit être déduit à partir des mesures des indicateurs.

Cependant, si des données liées aux événements sont connues ainsi que le temps où ces données ont été prises, on peut, à partir des mesures des indicateurs et les temps où ces mesures ont été prises, déduire l'état de l'équipement pour certaines de ces observations

Cette procédure permettrait donc d'avoir un historique d'observations et leur état correspondant. Cet historique d'observations est indispensable pour la génération de patrons pour l'algorithme LAD. Cette étape du LAD connue sous le nom d'apprentissage, consiste à extraire des relations à partir des données de l'historique sous forme de patrons logiques.

Dans la section suivante, les étapes du LAD sont décrites et expliquées à l'aide d'exemples. Il s'agit de la méthodologie utilisée pour implémenter le logiciel LAD pour la maintenance conditionnelle dans le cadre de cette recherche.

2.1 Méthodologie du LAD

Le but du LAD appliqué à la maintenance conditionnelle est de pouvoir caractériser l'état de la machine afin de planifier des actions de maintenance. On s'intéresse plus particulièrement à l'état où une machine peut développer une défaillance potentielle et que le risque de panne est accru, qu'il s'agisse d'une panne qui interrompe la fonction de la machine (panne évidente) où qui diminue sa performance (panne cachée).

Lorsque l'on identifie qu'une machine passe de l'état de fonctionnement normal vers un état de défaillance potentielle, des actions de maintenance peuvent être planifiées pour éviter qu'une panne ne se produise, éliminant ainsi les coûts potentiels engendrés par le temps d'arrêt et les coûts supérieurs d'une intervention de maintenance non planifiée.

Le LAD se sert des indicateurs de la machine telle que la concentration de métaux dans les lubrifiants, l'amplitude et les fréquences d'oscillation entre autres pour caractériser l'état de la machine à un instant précis. Un vecteur de mesures des indicateurs de la machine est appelé observation de dimension n si elle contient n indicateurs.

Le LAD analyse l'historique de données pour découvrir des connaissances ou des informations structurelles sous forme de patrons logiques. Ces derniers sont des combinaisons des valeurs des indicateurs qui se présentent dans un seul des états de la machine. C'est ainsi qu'un patron caractéristique positif est un vecteur de valeurs des indicateurs qui caractérise au moins une observation lorsque la machine était dans l'état que l'on appelle positif (état dégradé de la machine selon la convention adoptée dans

cette recherche) et ne caractérise aucune observation dont l'état est négatif ou de fonctionnement normal de la machine. La réciproque est vraie pour les patrons négatifs.

Les observations qui servent à générer ces patrons forment un groupement appelé ensemble d'apprentissage et elles sont aussi connues comme historique car ce sont des mesures prises dans le passé. L'algorithme du LAD est un algorithme qui utilise l'apprentissage supervisé car les données de l'historique contiennent non seulement les mesures des indicateurs mais également l'état associé de l'observation. Pour les nouvelles observations, par contre, on ne connaît que les mesures des indicateurs et le but du LAD est de déterminer l'état dans lequel ces observations sont prises.

Le résultat (l'état) peut correspondre à un état de fonctionnement normal, à l'état de défaillance potentielle ou à plusieurs états spécifiques aux types de panne. Bien qu'à la base, le LAD soit une méthode qui ne fonctionne que pour des données appartenant à un de deux états, il existe cependant des façons de l'adapter aux cas qui présentent davantage d'états (Hammer, P.L., 2000).

Dans ce mémoire on se limite à traiter le cas de deux états : l'état de fonctionnement normal et l'état de défaillance potentielle.

Les mesures des indicateurs peuvent être des nombres réels, nominaux ou des valeurs discrètes. Dans tous les cas (sauf si chaque indicateur ne prend que deux valeurs possibles, c'est-à-dire qu'il s'agit d'un indicateur qui est déjà en format binaire), il faut procéder à une étape de conversion binaire au préalable.

À l'aide de patrons obtenus lors de l'étape d'apprentissage, on peut classer les nouvelles observations dans un de deux états selon les patrons qui couvrent (valident) ces observations.

Les patrons, qui caractérisent l'information structurée extraite de l'ensemble d'apprentissage, permettent de construire une fonction pseudo booléenne de classification qui assigne des poids à chaque patron pour donner de l'importance à certains de ces derniers dans la classification. La classification consiste donc à évaluer la fonction pseudo-booléenne pour chaque nouvelle observation.

Le LAD que nous avons appliqué dans ce travail se fait donc en quatre grandes étapes :

1) La conversion binaire de données 2) La génération de patrons 3) La formation de la fonction discriminante, et finalement 4) La classification des nouvelles observations.

Afin d'illustrer les différentes étapes du LAD, nous aborderons un exemple qui étudie la prescription de verres de contact selon les caractéristiques du patient (Cendrowska, J., 1987). Nous avons choisi cet exemple à cause de sa simplicité, et aussi parce qu'il s'agit d'un exemple qui permet d'illustrer le pouvoir explicatif du LAD, c'est-à-dire de comprendre les connaissances extraites à partir de l'historique de données, et comment elles sont utilisées pour classer de nouvelles données. Dans les chapitres suivants, nous introduirons les applications dans le domaine de la maintenance conditionnelle.

Il s'agit de 24 observations qui consistent en quatre indicateurs nominaux et l'état. Il existe 3 états possibles, qui correspondent au type de prescription recommandée, soit : verres de contact rigides (1), verres de contact souples (2) et pas de verres de contact

(3). Voir le Tableau 2.1 Observations de l'étude de prescription de verres de contact (Cendrowska, J., 1987)

Tableau 2.1 Observations de l'étude de prescription de verres de contact (Cendrowska, J., 1987)

# Obs.	État	Âge	Type de verres	Astigmatique	Production de larmes
1	3	1	1	1	1
2	2	1	1	1	2
3	3	1	1	2	1
4	1	1	1	2	2
5	3	1	2	1	1
6	2	1	2	1	2
7	3	1	2	2	1
8	1	1	2	2	2
9	3	2	1	1	1
10	2	2	1	1	2
11	3	2	1	2	1
12	1	2	1	2	2
13	3	2	2	1	1
14	2	2	2	1	2
15	3	2	2	2	1
16	3	2	2	2	2
17	3	3	1	1	1
18	3	3	1	1	2
19	3	3	1	2	1
20	1	3	1	2	2
21	2	3	2	1	2
22	2	3	2	1	2
23	3	3	2	2	1
24	3	3	2	2	2

Comme dans le cas de cette recherche du LAD appliquée à la maintenance conditionnelle, nous ne faisons la distinction que de deux états et nous avons réduit les observations à deux états : verres de contact (0), pas de verres de contact (1).

Tableau 2.2 Observations de l'étude de prescription de verres de contact à deux états, basé sur (Cendrowska, J., 1987)

# Obs.	État	Âge	Type de verres	Astigmatique	Production de larmes
1	1	1	1	1	1
2	0	1	1	1	2
3	1	1	1	2	1
4	0	1	1	2	2
5	1	1	2	1	1
6	0	1	2	1	2
7	1	1	2	2	1
8	0	1	2	2	2
9	1	2	1	1	1
10	0	2	1	1	2
11	1	2	1	2	1
12	0	2	1	2	2
13	1	2	2	1	1
14	0	2	2	1	2
15	1	2	2	2	1
16	1	2	2	2	2
17	1	3	1	1	1
18	1	3	1	1	2
19	1	3	1	2	1
20	0	3	1	2	2
21	0	3	2	1	2
22	0	3	2	1	2
23	1	3	2	2	1
24	1	3	2	2	2

Les quatre indicateurs sont : l'âge du patient, le type de verre, l'astigmatisme et la production de larmes.

Les valeurs nominales des indicateurs sont :

Âge du patient : Jeune (1), pré-presbytique (2) et presbytique (3).

Type de verre : Myope (1), hypermétrope (2).

Astigmatique : Oui (1), non(2).

Production de larmes : Réduite (1) ou normale (2).

2.1.1 La conversion binaire de données

En général, les valeurs des indicateurs peuvent être nominales, tels que des entiers ou des valeurs numériques (entiers réels), ainsi que des énumérations (i.e. de couleur : jaune, rouge, bleu).

Cependant, le LAD requiert que les valeurs des indicateurs soient des valeurs binaires (1 ou 0). La conversion binaire (chiffage binaire) est donc nécessaire pour convertir ces valeurs en valeurs binaires.

Une conséquence importante de la conversion binaire est le fait que l'interprétation des connaissances extraites sous forme de patrons et de résultats n'est pas directe. En effet, lorsque l'on effectue une conversion binaire, on traduit un indicateur original en un ou plusieurs indicateurs binaires. Lorsque les patrons sont générés, ils se constituent de ces indicateurs binaires et la relation avec les indicateurs d'origine est perdue.

Si le but dans un contexte donné est non seulement d'arriver à un résultat donné mais aussi de comprendre comment un résultat donné a été produit, il est nécessaire de pouvoir convertir non seulement des données nominales et numériques en données binaires, mais aussi l'inverse. C'est-à-dire être capable de traduire les patrons extraits

(formés d'indicateurs binaires) dans le format des indicateurs originaux. Cette fonctionnalité n'étant pas absolument fondamentale au fonctionnement de base de l'algorithme du LAD, n'a pas été implémentée dans le logiciel LAD-CBM, mais elle est fortement recommandée pour des versions ultérieures du logiciel.

Comme l'exemple de prescription de verres de contact ne contient que des données nominales, nous avons complété ce chapitre avec un autre exemple, afin de décrire la conversion binaire de données numériques.

Données nominales

Si les données d'origine sont des énumérations, il suffit d'assigner un nombre binaire au premier élément de l'énumération, d'en assigner le nombre binaire suivant au deuxième élément et ainsi de suite pour couvrir l'ensemble d'éléments.

Dans l'exemple de l'étude de prescription de verres de contact, nous avons trois valeurs possibles pour l'âge du patient (1, 2,3), deux valeurs possibles pour la prescription du verre (1,2), deux valeurs possibles pour l'astigmatisme (1,2) et deux valeurs possibles pour la production de larmes (1,2).

Pour représenter les trois valeurs possibles pour l'âge, il nous faut deux chiffres binaires, car 11 en base binaire représente 3 en base décimale, et les chiffres de 0 à 3 sont suffisants pour représenter 4 éléments. Pour les autres indicateurs qui sont déjà binaires

(1 ou 2), nous allons tout simplement les renommer en 0 et 1 respectivement. Voici une synthèse de la conversion binaire des indicateurs; les résultats de la conversion binaire de 24 observations se trouvent à l'Annexe A.

$$\begin{bmatrix} \text{Âge} \\ \text{jeune} \\ \text{pre - presbytique} \\ \text{presbytique} \end{bmatrix} = \begin{bmatrix} b0 & b1 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} \text{Type de verre} \\ \text{myope} \\ \text{hypermetrope} \end{bmatrix} = \begin{bmatrix} b2 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} \text{Astigmatique} \\ \text{oui} \\ \text{non} \end{bmatrix} = \begin{bmatrix} b3 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} \text{Production larmes} \\ \text{réduite} \\ \text{normale} \end{bmatrix} = \begin{bmatrix} b4 \\ 0 \\ 1 \end{bmatrix}$$

En général si k est le nombre de valeur différentes que peut prendre un indicateur (nombre d'éléments différents), et si $\lceil \log_2(k) \rceil$, est le plus petit entier supérieur (fonction plafond) au logarithme à la base 2 de k . $\lceil \log_2(k) \rceil$, représente le nombre d'indicateurs binaires nécessaires pour représenter ces k éléments (Williams H., 1999).

Dans l'exemple précédent où nous avons 3 valeurs possibles à encoder pour l'âge du patient, $\lceil \log_2(3) \rceil = \lceil 1.585 \rceil = 2$ est le nombre de chiffres binaires nécessaires pour encoder les trois différentes valeurs que peut prendre l'âge du patient.

Données numériques

Pour des cas où les valeurs sont entières ou continues il faut cependant adopter une procédure différente. Dans la vie courante, lorsqu'on mesure la température d'une personne par exemple, un thermomètre nous donne une mesure numérique. Cette mesure peut ensuite être comparée à certaines limites pour établir si la température est basse, normale ou haute. La conversion binaire de données continues ou entières suit la même logique.

Nous adoptons une technique qui se base sur le même principe pour définir des limites ou des points de coupure basés sur les valeurs de l'historique pour chaque indicateur de base. Ensuite, des indicateurs binaires sont assignés aux intervalles définis par ces points de coupure.

La première étape est celle d'ordonner les valeurs de chaque indicateur en ordre croissant et ce, pour l'ensemble d'observations de l'historique. Avec le vecteur ordonné des indicateurs, un point de coupure est défini pour chaque paire de valeur d'indicateurs qui se succèdent et sont associés à des états différents.

Ensuite, lorsque l'on veut faire la conversion binaire, on compare la valeur d'un indicateur et les différents points de coupure établis pour l'indicateur en question.

Afin d'illustrer la conversion binaire, nous allons traiter l'exemple des observations d'un roulement mécanique de l'expérience décrite en (Sun Q., et al., 2004). Cet exemple sera traité au complet à la section 4.1.

Voici les six indicateurs, les mesures étant des valeurs réelles :

Tableau 2.3 Observations d'un roulement mécanique. (Sun Q. et al, 2004)

État	# Mesure	Rv	Pk	Cf	Kv	Cl	If
Défectueux	1	7.8491	22.987	2.9301	3.4406	6.3869	3.7442
	2	4.9825	12.5721	2.5233	3.1313	5.6337	3.2916
	3	6.0294	15.125	2.5085	2.5077	5.3118	3.1450
	4	10.3168	31.1830	3.0225	3.5191	6.9424	3.9892
Normal	5	1.2348	3.8428	3.1129	3.6318	7.814	4.2400
	6	0.8953	2.6677	2.9707	3.2602	6.7993	3.8483
	7	0.9193	2.8486	3.0987	3.4304	7.2580	4.0608
	8	1.069	3.5156	3.2887	3.4158	8.5502	4.5622

Nous devons en premier lieu mettre les données de chaque colonne en ordre croissant tout en retenant l'état auquel appartient chaque observation. Ici Pos dénote l'état positif ou défectueux, et Neg dénote l'état Négatif ou normal.

Tableau 2.4 Roulement mécanique: Indicateurs ordonnancés et états associés

R _v		P _k		C _f		K _v		C _l		I _f	
État	Valeur	État	Valeur	État	Valeur	État	Valeur	État	Valeur	État	Valeur
Neg	0.8953	Neg	2.6677	Pos	2.5085	Pos	2.5077	Pos	5.3118	Pos	3.1450
Neg	0.9193	Neg	2.8486	Pos	2.5233	Pos	<u>3.1313</u>	Pos	5.6337	Pos	3.2916
Neg	1.069	Neg	3.5156	Pos	<u>2.9301</u>	Neg	<u>3.2602</u>	Pos	<u>6.3869</u>	Pos	<u>3.7442</u>
Neg	<u>1.2348</u>	Neg	<u>3.8428</u>	Neg	<u>2.9707</u>	Neg	3.4158	Neg	<u>6.7993</u>	Neg	<u>3.8483</u>
Pos	<u>4.9825</u>	Pos	<u>12.5721</u>	Pos	<u>3.0225</u>	Neg	<u>3.4304</u>	Pos	<u>6.9424</u>	Pos	<u>3.9892</u>
Pos	6.0294	Pos	15.125	Neg	<u>3.0987</u>	Pos	<u>3.4406</u>	Neg	<u>7.2580</u>	Neg	<u>4.0608</u>
Pos	7.8491	Pos	22.987	Neg	3.1129	Pos	<u>3.5191</u>	Neg	7.814	Neg	4.2400
Pos	10.3168	Pos	31.1830	Neg	3.2887	Neg	<u>3.6318</u>	Neg	8.5502	Neg	4.5622

Si nous prenons l'indicateur R_v , il n'y a qu'une seule alternance d'état, soit de l'état négatif avec une valeur de $V_{s-1} = 1.2348$ vers l'état positif de valeur $V_s = 4.9285$. Il n'existe qu'un seul point de coupure que nous établissons à l'aide de la formule suivante :

$$P_c = \frac{1}{2} (V_{s-1} + V_s)$$

Équation 2.1 Point de coupure

Ici P_c est le point de coupure et V_{s-1} la valeur de l'indicateur dont l'observation est de l'état de départ et V_s est la valeur de l'indicateur dont l'observation présente le signe contraire.

Il y a deux types de variables booléennes qui seront utilisés pour l'assignation d'indicateurs binaires :

- Les variables de niveau sont associées à chaque point de coupure et décrivent si la valeur de l'indicateur original est supérieure ou inférieure au point de coupure.

$$V_i \geq P_{c_j}; B_{n_{ij}} = 1$$

$$V_i < P_{c_j}; B_{n_{ij}} = 0$$

Où : V_i est la valeur de l' $i^{\text{ème}}$ indicateur original.

P_{c_j} est le $j^{\text{ème}}$ point de coupure.

$B_{n_{ij}}$ est la valeur de l'indicateur booléen de niveau associé.

à l' $i^{\text{ème}}$ indicateur et le $j^{\text{ème}}$ point de coupure.

- Les variables d'intervalle sont assignées à chaque paire de points de coupure et décrivent si la valeur de l'indicateur original est dans l'intervalle défini par les points de coupure.

$$Pc_j \leq V_l < Pc_k \rightarrow Bi_{ljk} = 1; \forall j < k$$

$$V_l < Pc_j ; V_l > Pc_k \rightarrow Bi_{ljk} = 0; \forall j < k$$

Où : Pc_j est le point de coupure j .

V_l est la valeur de l'indicateur l .

Bi_{ljk} est la valeur de l'indicateur booléen d'intervalle associée à l'indicateur l et aux points de coupure j et k .

Dans l'exemple, il n'y a qu'un seul indicateur binaire pour la colonne R_v qui correspond à une variable booléenne de niveau associé au seul point de coupure, $P_c = \frac{1}{2} (1.2348 + 4.9285) = 3.08165$. Il n'y a pas d'indicateurs binaires liés aux variables d'intervalle car il n'existe qu'un seul point de coupure.

$$Bn_{11} = 1 \text{ si } ; V_1 \geq 3.08165$$

$$Bn_{11} = 0 \text{ si } ; V_1 < 3.08165$$

Où V_1 est R_v selon l'ordre d'indicateurs dans le Tableau 2.3 Observations d'un roulement mécanique. (Sun Q. et al, 2004)

Si nous prenons la colonne C_6 , il existe alors trois transitions : soit de 2.9301 (Pos) vers 2.9707 (Neg), de 2.9707 (Neg) vers 3.0225 (Pos) et de 3.0225 (Pos) vers 3.0987 (Neg).

Dans l'exemple du tableau précédent, il y a donc 3 variables booléennes de niveau pour les trois points de coupure correspondants (calculées comme dans le cas de la colonne R_v).

Les points de coupure sont :

$$P_{c1} = (2.9301 + 2.9707) / 2 = 2.9504$$

$$P_{c2} = (2.9707 + 3.0225) / 2 = 2.9966$$

$$P_{c3} = (3.0225 + 3.0987) = 3.0606.$$

Voici les variables de niveau associées :

$$B_{n31} = 1 \text{ si } ; V_3 \geq 2.9504$$

$$B_{n31} = 0 \text{ si } ; V_3 < 2.9504$$

$$B_{n32} = 1 \text{ si } ; V_3 \geq 2.9966$$

$$B_{n32} = 0 \text{ si } ; V_3 < 2.9966$$

$$B_{n33} = 1 \text{ si } ; V_3 \geq 3.0606$$

$$B_{n33} = 0 \text{ si } ; V_3 < 3.0606$$

Où V_3 est C_f selon l'ordre d'indicateurs dans le Tableau 2.3 Observations d'un roulement mécanique. (Sun Q. et al, 2004)

Dans le cas des variables booléennes d'intervalle, voici tous les intervalles possibles selon les points de coupure déterminés soit: entre 2.9504 et 2.9966, entre 2.9504 et 3.0606 et entre 2.9966 et 3.0606.

Au total, il y a trois intervalles, cependant certains de ces intervalles sont redondants. Par exemple, si un point se trouve dans l'intervalle compris entre 2.9966 et 3.0606 ou entre 2.9504 et 2.9966, il sera toujours dans l'intervalle 2.9504 et 3.0606. De la même façon, si un point n'est pas dans l'intervalle entre 2.9966 et 3.0606 ni entre 2.9504 et 2.9966, dans tous les cas, ce point ne se trouve pas non plus entre 2.9504 et 3.0606.

Donc, nous ne conservons que les intervalles strictement nécessaires pour décrire le phénomène, soit ceux définis entre 2.9504 et 2.9966, et entre 2.9966 et 3.0606.

En règle générale, si nous avons une liste ordonnancée de points de coupure en ordre croissant, nous ne gardons que les intervalles définis par chaque paire de points de coupure qui se succèdent, car tous les autres intervalles possibles sont déjà contenus dans les intervalles considérés. Cette approche aide grandement à réduire le nombre d'indicateurs binaires et diminue aussi la complexité de calculs et le temps d'exécution de la phase de génération de patrons sans nuire à la performance de la solution. Elle a été utilisée dans l'implémentation logicielle de l'algorithme LAD que nous détaillerons au chapitre suivant.

C'est ainsi que dans l'exemple, nous aurons trois indicateurs liés aux variables de niveau pour l'indicateur Cf et deux indicateurs binaires liés aux variables d'intervalle.

Indicateurs binaires liés aux variables de niveau :

$$B_{n31} = 1 \text{ si } ; V_3 \geq 2.9504$$

$$B_{n31} = 0 \text{ si } ; V_3 < 2.9504$$

$$B_{n32} = 1 \text{ si } ; V_3 \geq 2.9966$$

$$B_{n32} = 0 \text{ si } ; V_3 < 2.9966$$

$$B_{n33} = 1 \text{ si } ; V_3 \geq 3.0606$$

$$B_{n33} = 0 \text{ si } ; V_3 < 3.0606$$

Indicateurs binaires liés aux variables d'intervalle :

$$B_{i312} = 1 \text{ si } 2.9504 \leq V_3 < 2.9966 \text{ sinon } B_{i312} = 0$$

$$B_{i323} = 1 \text{ si } 2.9966 \leq V_3 < 3.0606 \text{ sinon } B_{i323} = 0$$

Il faut procéder de façon identique pour chaque indicateur qui compose l'observation pour obtenir des indicateurs binaires correspondants. À continuation nous présentons la synthèse des règles de conversion binaire pour chaque indicateur :

Rv, 1 point de coupure son indicateur binaire associé, soit :

$$Bn_{11} = 1 \text{ si } ; V_1 \geq 3.10865, \text{ sinon, } Bn_{11} = 0$$

Pk, 1 point de coupure son indicateur binaire associé, soit :

$$Bn_{21} = 1 \text{ si } ; V_2 \geq 8.20745, \text{ sinon, } Bn_{21} = 0$$

Cf, 3 points de coupure et les 5 indicateurs binaires associés, soit :

$$Bn_{31} = 1 \text{ si } ; V_3 \geq 2.9504, \text{ sinon, } Bn_{31} = 0$$

$$Bn_{32} = 1 \text{ si } ; V_3 \geq 2.9966, \text{ sinon, } Bn_{32} = 0$$

$$Bn_{33} = 1 \text{ si } ; V_3 \geq 3.0606, \text{ sinon, } Bn_{33} = 0$$

$$Bi_{312} = 1 \text{ si } 2.9504 \leq V_3 < 2.9966 \text{ sinon } Bi_{312} = 0$$

$$Bi_{323} = 1 \text{ si } 2.9966 \leq V_3 < 3.0606 \text{ sinon } Bi_{323} = 0$$

Kv, 3 points de coupure et les 5 indicateurs binaires associés, soit :

$$Bn_{41} = 1 \text{ si } ; V_4 \geq 3.19575, \text{ sinon, } Bn_{41} = 0$$

$$Bn_{42} = 1 \text{ si } ; V_4 \geq 3.4355, \text{ sinon, } Bn_{42} = 0$$

$$Bn_{43} = 1 \text{ si } ; V_4 \geq 3.5745, \text{ sinon, } Bn_{43} = 0$$

$$Bi_{412} = 1 \text{ si } 3.19575 \leq V_4 < 3.4355 \text{ sinon } Bi_{412} = 0$$

$$Bi_{423} = 1 \text{ si } 3.4355 \leq V_4 < 3.5745 \text{ sinon } Bi_{423} = 0$$

Cl, 3 points de coupure et les 5 indicateurs binaires associés, soit :

$$Bn_{51} = 1 \text{ si } ; V_5 \geq 6.5931, \text{ sinon, } Bn_{51} = 0$$

$$Bn_{52} = 1 \text{ si } ; V_5 \geq 6.87085, \text{ sinon, } Bn_{52} = 0$$

$$Bn_{53} = 1 \text{ si } ; V_5 \geq 7.1002, \text{ sinon, } Bn_{53} = 0$$

$$Bi_{512} = 1 \text{ si } 6.5931 \leq V_5 < 6.87085 \text{ sinon } Bi_{512} = 0$$

$$Bi_{523} = 1 \text{ si } 6.87085 \leq V_5 < 7.1002 \text{ sinon } Bi_{523} = 0$$

If, 3 points de coupure et 5 indicateurs binaires associés, soit :

$$Bn_{61} = 1 \text{ si } ; V_6 \geq 3.79625, \text{ sinon, } Bn_{61} = 0$$

$$Bn_{62} = 1 \text{ si } ; V_6 \geq 3.91875, \text{ sinon, } Bn_{62} = 0$$

$$Bn_{63} = 1 \text{ si } ; V_6 \geq 4.025, \text{ sinon, } Bn_{63} = 0$$

$$Bi_{612} = 1 \text{ si } 3.79625 \leq V_6 < 3.91875 \text{ sinon } Bi_{612} = 0$$

$$Bi_{623} = 1 \text{ si } 3.91875 \leq V_6 < 4.025 \text{ sinon } Bi_{623} = 0$$

Le Tableau 2.5 illustre les données d'origines converties en format binaires :

Tableau 2.5 Roulement mécanique: Données en format binaire

#M	E	Rv	Pk	Cf						Kv					Cl					If				
		Bn ₃₁	Bn ₃₂	Bn ₃₃	Bn ₃₄	Bn ₃₅	Bn ₃₆	Bn ₃₇	Bn ₃₈	Bn ₄₁	Bn ₄₂	Bn ₄₃	Bn ₄₄	Bn ₄₅	Bn ₅₁	Bn ₅₂	Bn ₅₃	Bn ₅₄	Bn ₅₅	Bn ₆₁	Bn ₆₂	Bn ₆₃	Bn ₆₄	Bn ₆₅
1	D	1	1	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
2	D	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	D	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	D	1	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0	1	1
5	N	0	0	1	1	1	0	0	1	1	1	0	0	1	1	1	1	0	0	1	1	1	0	0
6	N	0	0	1	0	0	1	0	1	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0
7	N	0	0	1	1	1	0	0	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	0
8	N	0	0	1	1	1	0	0	1	0	0	1	0	1	1	1	1	0	0	1	1	1	0	0

Légende : Colonne # M : # Mesure. Colonne E : Colonne État avec les valeurs suivantes D : État Défectueux ; N : État Normal

L'étape qui suit la conversion binaire est l'extraction de connaissances ou plus spécifiquement, la génération de patrons.

2.1.2 La génération de patrons

Les données à analyser font partie d'une archive historique d'observations en format binaire (que l'on dénote E) à l'étape précédente. Chaque observation est un vecteur de mesures de n indicateurs, auquel se joigne l'indicateur d'état. Si une observation est associée à l'état positif (défaillance potentielle), elle fait partie de l'ensemble E+. La réciproque est vraie pour les observations négatives (état normal) et l'état E-.

Par convention, lorsque l'état associé à une observation est positif (défaillance potentielle) la variable booléenne d'état associé est égale à 0, alors que cette variable booléenne d'état est égale à 1 pour les observations négatives (fonctionnement normal).

Avant de décrire la génération de patrons en détail, il est important de connaître certains concepts :

Dans l'algèbre booléenne, un littéral est défini comme une variable booléenne ou sa négation, alors qu'un terme est une combinaison de littéraux. Le nombre de littéraux d'un terme est son degré. Un terme couvre une observation, si le résultat du terme évalué à cette observation est positif.

Un terme candidat (ou candidat tout simplement) est un terme qui couvre au moins une observation positive et au moins une observation négative.

Un terme caractéristique (minterm) d'une observation de degré n , est le seul terme de degré n qui couvre cette observation.

Il est également important de prendre en considération la notation suivante qui sera utilisée dans ce travail. Si nous ordonnons les indicateurs booléens en ordre, nous avons que la variable booléenne x_1 dénote que le premier indicateur a une valeur booléenne de 1, alors $\overline{x_1}$ dénote que le premier indicateur a une valeur booléenne de 0.

En synthèse, tout terme d'un degré donné i peut appartenir à une des classes suivantes :

P_i^+ , sont les termes de degré i qui couvrent au moins une observation appartenant à E^+ et aucune observation de l'historique appartenant à E^- . On les nomme patrons positifs.

P_i^- , sont les termes de degré i qui couvrent au moins une observation appartenant à E^- et aucune observation de l'historique appartenant à E^+ . On les nomme patrons négatifs.

C_i , sont des termes qui couvrent au moins une observation négative et une observation positive. On les nomme termes candidats de degré i .

L_i sont tous les autres termes de degré i qui sont exclus des itérations ultérieures.

Un patron est prime s'il est minimal, c'est-à-dire si tous les termes dérivés en éliminant des littéraux ne sont pas des patrons. Il est important d'introduire quelques concepts liés à l'algorithme de génération de patrons.

Un terme candidat (ou candidat tout simplement) est un terme qui couvre au moins une observation positive et au moins une observation négative. C'est justement en augmentant le nombre d'indicateurs binaires inclus dans un candidat que l'on peut faire en sorte que le candidat ne couvre que des observations appartenant à un seul état. Si le terme résultant de l'ajout de nouveaux indicateurs binaires, ne couvre que des observations positives, le terme devient un patron positif. Si le terme résultant ne contient que des observations négatives, le terme devient un patron négatif.

Exemple :

Nous avons l'observation $(1,1,0,1,0)^-$, et l'observation $(1,0,1,1,0)^+$, où le signe représente l'état associé à l'observation. Le terme caractéristique de $(1,1,0,1,0)$ est $x_1 x_2 \overline{x_3} x_4 \overline{x_5}$ et le terme caractéristique de $(1,0,1,1,0)$ est $x_1 \overline{x_2} x_3 x_4 \overline{x_5}$

$x_1 \overline{x_3} \overline{x_5}$ est un patron négatif, il couvre la première observation (négative) et ne couvre aucune observation positive. $x_1 \overline{x_3} \overline{x_5}_{(1,1,0,1,0)}$ est vrai (c'est-à-dire le patron évalué au

point 1,1,0,1,0 est vrai) car dans l'observation (1,1,0,1,0), $x_1 = 1$, $x_3 = 0$, $x_5 = 0$. Cependant il ne s'agit pas d'un patron prime négatif car si nous enlevons le littéral $\overline{x_5}$, le terme résultant $x_1 \overline{x_3}$ est toujours un patron négatif car il couvre l'observation négative mais ne couvre pas l'observation positive.

De plus, $x_1 \overline{x_3}$ n'est pas un patron prime négatif, car si nous enlevons le littéral x_1 , le terme $\overline{x_3}$ est un patron négatif. Le terme $\overline{x_3}$ est un patron prime négatif (nous ne pouvons plus enlever de littéraux).

Si une nouvelle observation est couverte par un patron positif, ceci est une évidence que l'observation appartient à E^+ . La réciproque est vraie pour les patrons négatifs, et l'appartenance à E^- . Il faut noter qu'une nouvelle observation peut être couverte en même temps par des patrons positifs et des patrons négatifs.

Cependant, nous pouvons nous rendre compte que la croissance du nombre de termes générés explose (croissance exponentielle). La notation $\binom{n}{d}$ représente le nombre de façons différentes que d éléments peuvent être choisis parmi n éléments. Cependant, comme chaque littéral peut être soit positif ou négatif alors $2^d \times \binom{n}{d}$, représente le nombre de termes différents de degré d qui peuvent être formés à partir de n variables booléennes.

$$\binom{n}{d} = \frac{n!}{d! (n-d)!}$$

Équation 2.2 Nombre de combinaisons différentes de d éléments parmi n

Avec, l'Équation 2.2 nous avons que le nombre de termes générés lorsque le degré maximal de patrons générés est d , est la somme des termes de toutes les étapes antérieures qui ont généré de termes de degré 1, de degré 2 et ainsi successivement jusqu'à arriver au degré d , soit :

$$\sum_{i=1}^d 2^i \times \binom{n}{i} = \sum_{i=1}^d 2^i \times \frac{n!}{i! (n-i)!}$$

Équation 2.3 Nombre de termes générés jusqu'au degré d

En faisant varier le degré maximal de patrons générés d , nous avons produit le Tableau 2.6 Synthèse du nombre de termes vs degré maximal de patrons générés qui illustre le phénomène de l'explosion combinatoire pour des observations à 20 indicateurs booléens.

Tableau 2.6 Synthèse du nombre de termes vs degré maximal de patrons générés

Degré maximal de patrons générés	Nombre de termes différents
3	9 920
4	87 440
5	583 568
6	3 064 208
7	12 986 768
8	45 235 088
9	131 230 608
10	320 420 752

Comme le nombre de termes que l'on doit analyser croît très rapidement même lorsque le nombre d'indicateurs est restreint, on doit procéder de la façon suivante : pour des observations à n indicateurs, l'algorithme de génération de patrons consiste en une approche ascendante de génération de patrons de petit degré suivie d'une approche descendante (on commence avec de patrons de haut degré), pour générer des patrons pour couvrir les observations qui ne sont pas couvertes par les patrons de petit degré.

Cette méthode réduit grandement le temps d'exécution et les ressources requises pour effectuer ce type de calcul sans pour autant sacrifier la qualité de patrons générés mesurée en terme de simplicité (patrons de petit degré) et nombre d'observations couvertes. Dans les paragraphes suivants, nous décrirons l'algorithme d'obtention des patrons positifs, celui de l'obtention de patrons négatifs étant son analogue.

Pour les termes de degré 1, on génère $2 \times n$ termes qui représentent chaque indicateur ainsi que sa négation. On analyse s'il s'agit d'un terme appartenant à P_1^+ , à C_1 ou s'il faut l'exclure des itérations suivantes, termes appartenant à L_1 . On ne retient que la liste de termes appartenant à P_1^+ , et C_1 .

C_1 sert de base pour la génération de termes de degré 2, car tout patron prime d'un degré supérieur est en réalité un candidat de degré inférieur auquel on a rajouté un littéral d'ordre lexicographique supérieur aux littéraux contenus dans le candidat.

L'ordre lexicographique permet de ne rajouter aux candidats que des littéraux qui n'ont pas été considérés, car autrement on répéterait l'analyse de certains termes. Pour l'exemple de la prescription de verres de contact, si nous tenons compte des

observations en format binaire (Voir Annexe A) , nous avons les indicateurs binaires suivants b_0, b_1, b_2, b_3, b_4 et l'ordre lexicographique s'établit comme suit $b_0 < \overline{b_0} < b_1 < \overline{b_1} < b_2 < \overline{b_2} < b_3 < \overline{b_3} < b_4 < \overline{b_4}$.

Pour générer les termes de degré 2 à être considérés, on part des termes candidats du degré immédiatement inférieur, dans ce cas l'ensemble de candidats C_1 et on doit connaître le littéral du plus grand d'ordre lexicographique contenu dans chaque candidat, on le nomme, $\max\text{Lit}(C_{1i})$, c'est-à-dire le littéral du plus grand ordre lexicographique du candidat i appartenant à l'ensemble de candidats de degré 1, C_1 . Chaque nouveau terme à être considéré sera le produit de générer toutes les combinaisons possibles qui résultent de rajouter un littéral (à la fois) qui est supérieur en ordre lexicographique à $\max\text{Lit}(C_{1i})$, et ce, pour chaque candidat i appartenant à C_1 .

Par la suite, chaque terme généré (de degré 2 dans notre cas) est classifié dans une des trois catégories : patrons primes, termes candidats et termes à exclure. On retient les P_2^+ , et les termes candidats de C_2 qui serviront de base de génération de termes pour le cycle suivant de degré 3 . Le restant des termes sont ignorés (L_2). Il est important de noter que l'on considère seulement les termes candidats auxquels on peut rajouter des littéraux d'ordre lexicographique supérieur à $\max\text{Lit}(C_i)$, et qu'une fois l'ajout effectué, ils présentent la propriété sur les sous termes de degré $i-1$ contenus dans les candidats de degré i :

Tous les soustermes de degré

$i - 1$ contenus dans un terme candidat de degré i (C_i) retenu, doivent être
des candidats contenus en C_{i-1} .

Cette propriété permet d'exclure la génération de candidats qui contiennent de sous-termes qui pourraient être des patrons primes de degré inférieur. Voir exemple dans les pages suivantes. (Hammer P.L, 2000).

On procède selon cet algorithme itératif jusqu'à avoir atteint le degré d (spécifié par l'utilisateur), jusqu'auquel on génère des patrons selon l'approche ascendante. Ce degré limite d , est spécifique au problème traité et dépend de la complexité du problème, c'est-à-dire du nombre d'indicateurs et du nombre d'observations.

Par la suite, on extrait les observations qui ne sont couvertes par aucun patron généré jusqu'ici, et on les utilise comme base pour l'approche descendante de génération de patrons. Cette étape consiste à commencer par les termes caractéristiques (voir définitions au début de la section) des observations non-couvertes et d'enlever un littéral à chaque itération.

À ce moment-ci, on analyse le terme obtenu à cette étape $d-1$, s'il s'agit d'un patron prime qui couvre encore l'observation correspondante, on enlève un autre littéral et on procède de la même façon dans l'itération $d-2$. Par contre, si le terme obtenu à $d-1$ ne couvre plus l'observation correspondante, ou n'est plus un patron prime, alors on déclare

le terme obtenu à $d-1$ comme un patron prime. On procède ainsi jusqu'à avoir couvert les observations manquantes.

Plus le degré d'un patron prime est petit, plus grand sera son pouvoir généralisateur, car il couvrira davantage d'observations de l'historique. Plus le degré d'un patron s'approche au nombre de littéraux des observations, plus il est spécifique à une observation donnée.

Si pour un problème donné, on trouve des patrons primes de degré 1, ceci veut dire que si l'on attribue un poids significatif à un de ces patrons (s'il couvre un grand nombre d'observations par exemple), un seul indicateur de la machine détermine en grand partie son état.

Par contre, dans le cas extrême, si le degré d'un patron est égal au nombre d'indicateurs des observations, on dit que le patron est plutôt le terme caractéristique d'une observation et qu'il ne couvre qu'une seule observation de l'historique.

Certains patrons générés se recoupent car ils couvrent certains points en commun. Dans la pratique, on élimine les patrons redondants qui sont ceux qui couvrent un sous-ensemble d'observations déjà couvertes par d'autres patrons primes.

Exemple:

Nous reprenons l'exemple traité dans la sous-section de conversion binaire, voir Tableau 2.2 Observations de l'étude de prescription de verres de contact à deux états, basé sur (Cendrowska, J., 1987).

Il faut définir au préalable un sous-ensemble d'apprentissage (historique) et un sous-ensemble de tests qui serviront à mettre à l'épreuve la fonction de classification qui sera développée à la sous-section suivante. Nous choisissons les observations appartenant aux ensembles d'apprentissage et de tests, de façon aléatoire mais en conservant la proportion d'observations appartenant à l'état 0 (environ 37%) et celles qui appartiennent à l'état 1 (63%) de l'ensemble de départ, nous définissons :

Les observations 1, 2, 5, 7, 9, 11, 13, 14, 15, 18, 21, 22 et 24 seront le sous-ensemble d'apprentissage qui servira pour générer les patrons.

Les observations 3, 4, 8, 10, 12, 16, 17, 19, 20, 23 seront le sous-ensemble de tests.

Lors de la première itération, on génère tous les termes possibles de degré 1 et on vérifie s'il y a des patrons primes. Dans notre cas, b_3 et $\overline{b_4}$ sont des patrons primes positifs car b_3 couvre les observations 7, 11, 15, 24, toutes des observations associées à l'état positif (leur état est 1), et ne couvre aucune observation négative. Le terme $\overline{b_4}$ couvre les observations 1, 5, 7, 9, 11, 13, 15, 21 et ne couvre aucune observation négative.

b_3 n'est pas un candidat faisant partie de C_1 car b_3 est déjà un patron prime positif et il en est de même pour $\overline{b_4}$ qui est aussi exclus des candidats. Au cycle de génération de termes de degré 2 et à chaque cycle suivant, on génère les termes à être considérés à partir des candidats du degré immédiatement inférieur en rajoutant un littéral d'ordre lexicographique supérieur à tous les littéraux présents dans le terme.

De plus, la propriété sur les soustermes de degré $i-1$ contenus dans les candidats de degré i , doit être respectée par les candidats C_2 . Par exemple, le terme $\overline{b_3} \overline{b_4}$, n'est pas retenu car un de sous-termes de degré i , soit $\overline{b_4}$ ne fait pas partie des candidats C_1 (Voir Tableau 2.7 Étude de prescription de verres de contact - Génération de patrons). Ce tableau, présente une synthèse de l'étape de génération de patrons (jusqu'au degré 2) pour l'exemple de prescription de verres de contact.

Tableau 2.7 Étude de prescription de verres de contact - Génération de patrons

Degré	Termes considérés	Patrons primes positifs	Patrons primes négatifs	Candidats
1	$b_0, \overline{b_0}, b_1, \overline{b_1}, b_2, \overline{b_2}, b_3, \overline{b_3}, b_4, \overline{b_4}$	$b_3, \overline{b_4}$		$b_0, \overline{b_0}, b_1, \overline{b_1}, b_2, \overline{b_2}, \overline{b_3}, b_4$
2	$b_0 b_1, b_0 \overline{b_1}, b_0 b_2, b_0 \overline{b_2}, b_0 \overline{b_3}, b_0 b_4, b_0 \overline{b_4}, \overline{b_0} b_1, \overline{b_0} \overline{b_1}, \overline{b_0} b_2, \overline{b_0} \overline{b_2}, \overline{b_0} \overline{b_3}, \overline{b_0} b_4, \overline{b_0} \overline{b_4}, b_1 b_2, b_1 \overline{b_2}, b_1 \overline{b_3}, b_1 b_4, b_1 \overline{b_4}, \overline{b_1} b_2, \overline{b_1} \overline{b_2}, \overline{b_1} \overline{b_3}, \overline{b_1} b_4, \overline{b_1} \overline{b_4}, b_2 \overline{b_3}, b_2 b_4, b_2 \overline{b_4}, \overline{b_2} \overline{b_3}, \overline{b_2} b_4, \overline{b_3} b_4, \overline{b_3} \overline{b_4}$	$b_0 \overline{b_2}, b_1 \overline{b_2}, \overline{b_1} b_2$	$\overline{b_0} b_4, \overline{b_1} b_4$	$b_0 b_1, b_0 \overline{b_1}, b_0 b_2, b_0 \overline{b_3}, b_0 b_4, \overline{b_0} b_1, \overline{b_0} \overline{b_1}, \overline{b_0} b_2, \overline{b_0} \overline{b_2}, \overline{b_0} \overline{b_3}, \overline{b_0} b_4, \overline{b_0} \overline{b_4}, b_1 b_2, b_1 \overline{b_3}, b_1 b_4, \overline{b_1} \overline{b_2}, \overline{b_1} \overline{b_3}, b_2 \overline{b_3}, b_2 b_4, \overline{b_2} \overline{b_3}, \overline{b_2} b_4, \overline{b_3} b_4$

Il est important de noter que dans le logiciel LAD-CBM, à l'étape de génération de patrons, on identifie aussi les observations couvertes par chaque patron pour éviter de refaire des évaluations à l'étape de la fonction discriminante.

Le Tableau 2.8, montre les observations couvertes par les patrons positifs et les patrons négatifs trouvés lorsque l'on fait la génération de patrons jusqu'au degré 5 en utilisant le logiciel LAD-CBM développé et décrit dans le Chapitre 4.

Tableau 2.8 Les patrons générés et leur couverture - Étude de prescription de verres de contact.

Nom du patron	Patron	Observations couvertes	# Observations couvertes
P_1	b_3	7,11,15,24	4
P_2	\bar{b}_4	1,5,9,13,15,21,7,11	8
P_3	$b_0\bar{b}_2$	18	1
P_4	\bar{b}_1b_2	5,7	2
P_5	$b_1\bar{b}_2$	9,18,11	3
N_1	\bar{b}_0b_4	2,14	2
N_2	\bar{b}_1b_4	2	1
N_3	$b_2\bar{b}_3b_4$	14,22	2

Dans le Tableau 2.8, les patrons nommés avec un grand p sont des patrons primes positifs, alors que ceux qui sont nommés avec un grand N sont des patrons primes négatifs. Même si on a indiqué au logiciel de chercher des patrons jusqu'au cinquième degré, il n'existe pas de patrons de degré plus grand que 3.

Il est intéressant de noter également au sujet du pouvoir explicatif de patrons, que le patron positif P_2 explique (couvre) à lui seul 72.7% des observations positives de l'historique, ce qui voudrait dire que \bar{b}_4 où le fait d'avoir un taux de production de

larmes réduites est un facteur déterminant pour ne pas formuler l'utilisation de verres de contact. De plus, la présence d'astigmatisme b_3 semble être aussi un facteur non-négligeable dans la non-prescription de verres de contact, cet indicateur à lui seul couvre 36.3% des observations positives de l'historique.

Quant à la prescription de verres de contact, le fait d'être hypermétrope b_2 mais non astigmatique \overline{b}_3 et d'avoir un taux de production de larmes normal b_4 couplé à l'indicateur d'être jeune, \overline{b}_0 , \overline{b}_1 sont des indices importants pour la prescription de verres de contact.

Le fait de pouvoir interpréter les connaissances extraites par l'algorithme est sans doute de grande valeur pour comprendre non seulement les données de l'historique et leur classification, mais également de comprendre comment le classificateur est arrivé à un résultat et ce, dans le domaine du problème grâce à l'interprétation des indicateurs présents dans les patrons (paragraphe précédent) à l'étape de classification de nouvelles observations.

On note que l'ensemble de patrons est complet, car les patrons générés couvrent toutes les observations (positives et négatives) de l'historique. Maintenant que nous avons les patrons et leur couverture, nous procédons à la détermination de la fonction discriminante qui sera la base pour la classification de nouvelles observations.

2.1.3 La fonction discriminante

Avec cet ensemble de patrons, le but est de trouver une fonction pseudo-booléenne appelée fonction discriminante, qui est une somme pondérée de patrons primes positifs et négatifs, et qui servira à classifier les nouvelles observations. Les termes correspondants aux patrons positifs seraient affectés des poids positifs, alors que les termes des patrons négatifs auraient de coefficients négatifs.

L'équation de la fonction discriminante est :

$$\Delta = \sum_{k=1}^r w_k^+ P_k + \sum_{l=1}^s w_l^- N_l$$

Équation 2.4 Fonction discriminante

Où :

$P_1, P_2 \dots P_r$ sont des patrons primes positifs générés à partir de l'ensemble d'apprentissage.

w_1^+ est le poids de P_1 , w_2^+ est le poids de P_2 , et w_r^+ est le poids de P_r .

$N_1, N_2 \dots N_s$ sont des patrons primes négatifs générés à partir de l'ensemble d'apprentissage.

w_1^- est le poids de N_1 , w_2^- est le poids de N_2 , et w_r^- est le poids de N_s .

Les poids assignés aux patrons signalent l'importance d'un patron déterminé dans le calcul de la fonction pseudo-booléenne. L'importance d'un patron peut être accordée entre autres, selon le nombre d'observations de l'historique qu'il couvre ou de la simplicité du patron en nombre de littéraux qui le composent. Dans le logiciel construit, nous avons calculé le poids en fonction du nombre d'observations de l'historique couvertes par chaque patron.

$$w_i^+ = \sum_{k=1}^n P_i(O_k)$$

Équation 2.5 Poids de patrons

Où

O_1 est la première observation de l'historique.

O_2 est la deuxième observation de l'historique.

O_n est la n^{ième} observation de l'historique.

$P_i(O_1)$ est l'évaluation de la première observation de l'historique par le patron prime positif i . $P_i(O_1)$ est égal à 1 si l'observation O_1 est couverte par le patron i , et égale à zéro autrement.

Pour que la fonction discriminante permette de classer des observations plus aisément selon une échelle standard, les poids sont normalisés afin qu'elle soit bornée entre -1 à 1.

Les poids normalisés doivent respecter l'équation suivante :

$$\sum_{k=1}^r w_{k\text{norm}}^+ = \sum_{l=1}^s w_{l\text{norm}}^- = 1$$

Équation 2.6 Somme de poids normalisés des patrons positifs et des patrons négatifs

La normalisation de poids se fait donc selon l'équation suivante :

$$w_{i\text{norm}}^+ = w_i^+ / \sum_{k=1}^r w_k^+$$

Équation 2.7 Poids normalisés de patrons positifs

Où

$w_{i\text{norm}}^+$ est le poids normalisé du i^{ème} patron prime positif

w_i^+ est le poids du i^{ème} patron prime positif avant la normalisation, établis selon

Équation 2.5 Poids de patrons

r est le nombre de patrons positifs

$$w_{j\text{norm}}^- = w_j^- / \sum_{l=1}^s w_l^-$$

$$w_{i\text{norm}}^+ = w_i^+ / \sum_{k=1}^r w_k^+$$

Équation 2.7 s'applique à la normalisation de patrons négatifs

$$w_{j\text{norm}}^- = w_j^- / \sum_{l=1}^s w_l^-$$

Équation 2.8 Poids normalisés de patrons négatifs

Où

$w_{j\text{norm}}^-$ est le poids normalisé du $i^{\text{ème}}$ patron prime négatifs.

w_j^- est le poids du $i^{\text{ème}}$ patron prime négatif avant la normalisation, établis selon

Équation 2.5 Poids de patrons

s est le nombre de patrons négatifs.

La fonction discriminante normalisée s'écrit :

$$\Delta_{norm} = \sum_{k=1}^r w_{k\text{norm}}^+ P_k + \sum_{l=1}^s w_{l\text{norm}}^- N_l$$

Équation 2.9 Fonction discriminante normalisée

Exemple :

Les poids se calculent selon l'Équation 2.5 Poids de patrons, pour notre exemple de prescription de verres de contact en nous référant au Tableau 2.8 Les patrons générés et

leur couverture, nous avons le calcul du poids pour le premier patron positif P_1 :

$$w_1^+ = \sum_{k=1}^n P_1(O_n) = 4$$

Comme nous l'avons mentionné auparavant, le logiciel LAD évalue les poids à la phase de génération de patrons pour éviter de refaire des évaluations des patrons vis-à-vis des observations, ainsi w_1^+ est tout simplement le nombre d'observations de l'historique

couvertes par le patron P_1 , c'est-à-dire la colonne de droite du Tableau 2.8, car pour toutes les autres observations $P_1(O_n)$ sera 0.

Après avoir calculé le poids de chaque patron positif, nous obtenons :

$$\sum_{k=1}^r w_k^+ = 4 + 8 + 1 + 2 + 3 = 18$$

$$\sum_{k=1}^s w_k^- = 2 + 1 + 2 = 5$$

Alors, le poids normalisé est calculé selon l'Équation 2.7 Poids normalisés de patrons positifs

$$w_{1\text{norm}}^+ = w_1^+ / \sum_{k=1}^r w_k^+ = \frac{4}{18} = 0.22$$

Le Tableau 2.9 montre les poids normalisés de chaque patron positif et négatif :

Tableau 2.9 Poids normalisés - Étude de prescription de verres de contact

Identifiant du poids	Valeur du poids
$w_{1\text{norm}}^+$	0.22
$w_{2\text{norm}}^+$	0.44
$w_{3\text{norm}}^+$	0.06
$w_{4\text{norm}}^+$	0.11
$w_{5\text{norm}}^+$	0.17
$w_{1\text{norm}}^-$	-0.4
$w_{2\text{norm}}^-$	-0.2
$w_{3\text{norm}}^-$	-0.4

$$\Delta_{\text{norm}} = \sum_{k=1}^r w_{k\text{norm}}^+ P_k + \sum_{l=1}^s w_{l\text{norm}}^- N_l$$

Équation 2.10 Les poids normalisés de patrons - Étude de prescription de verres de contact

$$\Delta_{norm} = (0.22 \times P_1 + 0.44 \times P_2 + 0.06 P_3 + 0.11 \times P_4 + 0.17 \times P_5) + (-0.4 \times N_1 + -0.2 \times N_2 + -0.4 \times N_3)$$

2.1.4 La classification

Lorsque la fonction est évaluée pour une nouvelle observation, on doit fixer une valeur limite pour la classifier comme positive ou négative. Une valeur de classification proche de -1 est un signe que l'observation est négative, c'est-à-dire prise d'une machine dans l'état normal selon notre convention, alors qu'une valeur proche de 1 signale que l'observation correspond à l'état positif, c'est-à-dire prise lorsque l'état de la machine est dans une défaillance potentielle.

Une valeur proche de 0, est dans une zone d'indécision où l'on ne peut pas se prononcer sur l'aspect positif ou négatif du résultat, c'est-à-dire l'état de la machine. .

Les valeurs limites de classification dépendent du type de problème traité et du profil de risque accepté. En maintenance, un faux positif est qualifié comme un avertissement ou une alarme qui était déclenchée pour signaler une condition anormale (défaillance potentielle) de la machine, alors qu'en réalité l'état de la machine est celui d'un

fonctionnement normal. C'est l'erreur type I. L'autre erreur possible est de ne pas détecter une défaillance potentielle. C'est l'erreur type II ou faux négatif.

Si le contexte est celui d'une centrale nucléaire, les conséquences d'une catastrophe font en sorte que le coût rattaché à des faux positifs est négligeable par rapport aux faux négatifs (à ne pas détecter un mal fonctionnement). Dans ce cas-ci, si la condition de défaillance potentielle est liée à l'état positif, le choix d'une valeur limite plus petite que 0 est un choix judicieux. Par exemple, si on décide de mettre le seuil de classification à -0.3 et que le plafond de classification pour les observations négatives est de -0.6, on voit que la zone de classification positive s'en trouve élargie, tel que montré dans la Figure 2.1 Zones de Classification des observations.

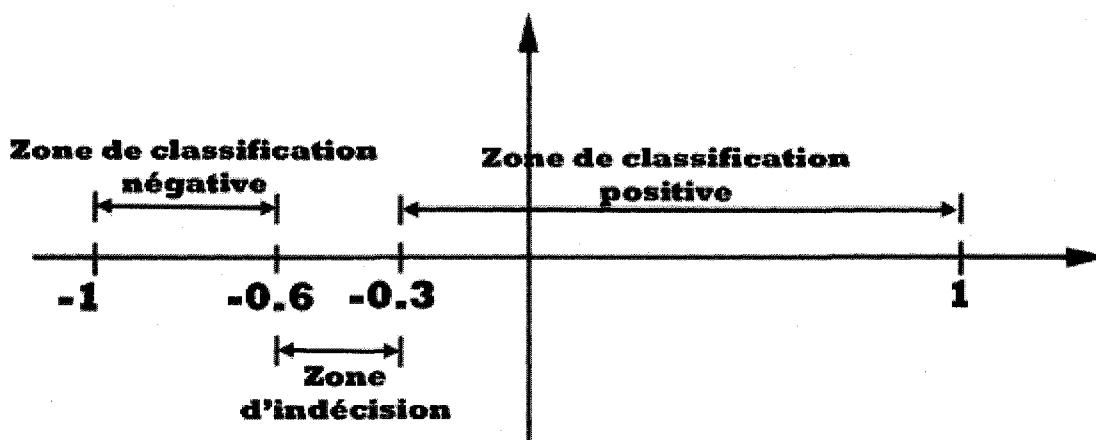


Figure 2.1 Zones de Classification des observations

Cependant, choisir une valeur exacte pour les limites de classification, est une tâche qui requiert de mettre l'algorithme à l'épreuve avec des nouvelles observations. Une fois

que l'état réel de la machine est connu pour un nombre considérable des observations de test, le taux d'erreur de classification pourrait être calculé.

Si ce taux est jugé inacceptable, les limites de classification pourraient être revues afin de classer la plupart des nouvelles données correctement. Pour éviter de changer les limites dans des cas particuliers, il faut soumettre des nouvelles observations à l'algorithme de classification afin de vérifier si le taux de classification erroné a en effet diminué. Sinon, il faudrait procéder de façon itérative jusqu'à ce qu'un bon compromis entre fausses alarmes et faux positifs soit trouvé tout en gardant le nombre d'observations qui tombent dans la zone d'indécision au minimum.

Dans le cas où il serait difficile ou impraticable de trouver des nouvelles observations de test et surtout de connaître leur état, une autre procédure existe. On utilise la fonction de classification (discriminante) sur les observations de l'historique et on compare les valeurs minimales et maximales de classification des observations négatives, ainsi que des observations positives pour décider quelle devrait être la valeur limite de classification.

Un graphique où on place ces valeurs minimales et maximales de classification des observations de l'historique (scores de référence) permet d'arriver à une décision de classification des nouvelles observations lorsque l'on compare le score de ces dernières avec les scores de référence, voir la. Figure 2.2. On définit un point neutre entre le score minimal des observations positives et le score maximal des observations négatives, et à partir de ce point neutre, on définit la zone d'indécision, la zone de classification

positive et la zone de classification négative. Pour voir un exemple de calcul, voir le Tableau 4.8 Exemple de calcul de la zone d'indécision et des zones de décision - Mécanisme de roulement

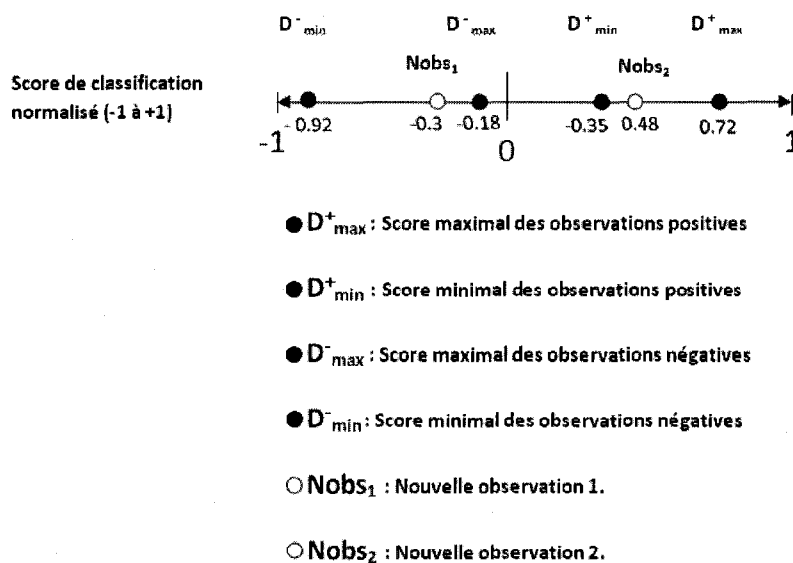


Figure 2.2 Approche graphique de classification par comparaison avec l'ensemble d'apprentissage

Exemple :

Nous avons l'ensemble de tests suivants (observations 3,4,6,8,10,12,16,17,19,20,23 de l'Annexe A), voir Tableau 2.10

Tableau 2.10 Ensemble de tests - Étude de prescription de verres de contact

# Obs.	État	Âge		Type verre	Astigmatique	Production de larmes
		b_0	b_1	b_2	b_3	b_4
3	1	0	0	0	1	0
4	0	0	0	0	1	1
6	1	0	0	1	0	1
8	0	0	0	1	1	1
10	0	0	1	0	0	1
12	0	0	1	0	1	1
16	1	0	1	1	1	1
17	1	1	1	0	0	0
19	1	1	1	0	1	0
20	0	1	1	0	1	1
23	1	1	1	1	1	0

La fonction discriminante est:

$$\Delta_{norm} = (0.22 \times P_1 + 0.44 \times P_2 + 0.06 P_3 + 0.11 \times P_4 + 0.17 \times P_5) + (-0.4 \times N_1 + -0.2 \times N_2 + -0.4 \times N_3)$$

Si nous écrivons la fonction discriminante en remplaçant les patrons primes par les termes qui les composent, nous avons :

$$\Delta_{norm} = (0.22 \times b_3 + 0.44 \times \bar{b}_4 + 0.06 b_0 \bar{b}_2 + 0.11 \times \bar{b}_1 b_2 + 0.17 \times b_1 \bar{b}_2) + (-0.4 \times \bar{b}_0 b_4 + -0.2 \times \bar{b}_1 b_4 + -0.4 \times b_2 \bar{b}_3 b_4)$$

Un exemple de calcul pour l'observation 3 est:

$$\begin{aligned}\Delta_{norm} &= (0.22 \times 1 + 0.44 \times 1 + 0.06 \times 0 + 0.11 \times 0 + 0.17 \times 0) + \\ &\quad (-0.4 \times 0 + -0.2 \times 0 + -0.4 \times 0) \\ &= 0.667\end{aligned}$$

Le résultat de l'observation est 0.66 sur notre échelle normalisée de -1 à 1. Mais il reste à définir quelle serait la limite de classification à partir de laquelle une observation serait positive.

Dans le cas le plus simple, on pourrait établir la limite à 0 et dire que tout résultat plus grand que 0 correspond à l'état positif, tandis que tout résultat plus petit que 0 est négatif.

Le résultat obtenu voudrait dire à priori que l'état de l'observation 3 est positif. Pour plus de certitude, on pourrait aborder l'approche graphique à la classification, voir Figure 2.2.

La synthèse de classification des observations de tests est la suivante :

Tableau 2.11 Résultats de classification - Étude de prescription de verres de contact

# Observation	Score de la fonction discriminante	État connu à priori
3	0.67	1
4	-0.38	0
6	-0.89	0
8	-0.27	0
10	-0.23	0
12	-0.01	0
16	-0.18	1
17	0.67	1

Tableau 2.11 Résultats de classification - Étude de prescription de verres de contact (suite et fin)

# Observation	Score de la fonction discriminante	État connu à priori
19	0.89	1
20	0.44	0
23	0.67	1

L'analyse nous permet de voir que les observations dont on connaissait l'état d'avance comme étant 1 sont toutes positives et suffisamment éloignées de la zone d'indécision (la valeur 0), à l'exception de l'observation numéro 16 qui a eu un score de classification de -0.18. Quant aux observations dont l'état connu à priori est de 0, elles sont toutes négatives et éloignées de la zone d'indécision, à l'exception de l'observation numéro 20 qui a un score de classification de 0.44 et l'observation numéro 12 qui est très proche de 0, soit d'une valeur de -0.01.

Si nous décidons que les observations dont le score de classification est supérieur à 0 seront classifiées comme étant positives et les observations dont le score de classification est inférieur à 0 sont négatives, le taux de classifications effectuées correctement est de 81.8% ou de neuf classifications correctes sur onze essais, voir Tableau 2.11.

Nous ne pouvons pas conclure que la performance de l'algorithme de classification soit de 81.8% pour le problème de la prescription de verres de contact, sans mener une étude plus exhaustive avec différents ensembles d'apprentissage et de données de test. De plus, le fait d'avoir fait une fusion des états nécessitant verres de contact rigides et verres

de contact souples en un seul état (verres de contact) afin de réduire le nombre d'états à deux (verres de contact, pas de verres de contact), a pu également jouer un effet sur l'exactitude des résultats.

Au chapitre 4, nous allons étudier deux applications du LAD au domaine de la maintenance conditionnelle, et dans un des cas, un plan d'expérience et une étude plus rigoureuse de la performance de classification seront effectués.

Pour mener des expériences répétées sur des ensembles d'observation de façon efficace, il est nécessaire de faire usage d'un outil qui automatise les calculs. Au chapitre suivant, nous allons décrire la structure du logiciel développé à cet effet et les interactions des différents modules qui composent le logiciel et sont responsables des différentes étapes du LAD.

CHAPITRE 3

CONCEPTION DU LOGICIEL

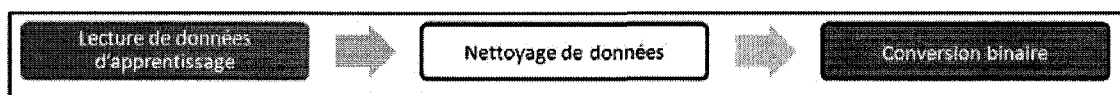
La motivation pour la construction du logiciel LAD-CBM (des sigles en anglais de l'analyse logique de données et de maintenance conditionnelle) découle des avantages offerts par un outil de support à la décision automatisée dans les milieux industriels.

Le langage de programmation C++ a été utilisé pour des raisons de performance ainsi que pour les caractéristique d'un langage orienté objets qui permettrait une plus grande modularité et faciliterait la maintenance du code existant ainsi que l'ajout des nouvelles fonctionnalités.

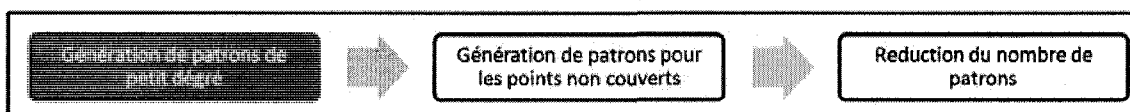
Le logiciel fournit une interface graphique qui permettrait d'entrer des paramètres importants lors des grandes étapes du LAD : dans la génération de patrons, la formation de la théorie de classification ou le processus de classification de nouvelles observations.

Voici un aperçu des modules qui composent le logiciel :

Préparation de données



Génération de patrons primes - Apprentissage



Construction du classificateur



Classification de nouvelles observations - Tests

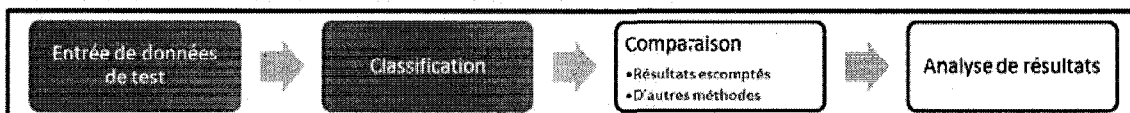


Figure 3.1 Modules du logiciel LAD-CBM

Comme le but de mon travail de maîtrise était celui de construire un cadre de travail pour l'application du LAD au CBM, seuls les principaux modules ont été implémentés (en bleu foncé dans le graphique précédent). Le critère de choix des modules a été de retenir ceux qui assurent le fonctionnement intégral de la démarche de classification, au dépens de ceux qui l'améliorent ou optimisent les résultats. Ces modules pourront faire l'objet d'une étude ultérieure.

Les modules qui ont été conçus sont les suivants :

Pour l'étape de la préparation de données, le module de lecture de données d'apprentissage a été implémenté.

Le module de binarisation a également été inclus, car pour l'analyse de résultats sur des ensembles de données avec un nombre d'indicateurs et d'observations considérable (plus de 10 indicateurs et autant d'observations et des essais répétitifs), la conversion binaire faite manuellement devient impraticable.

Pour l'étape de génération de patrons primes (apprentissage), le module de génération de patrons primes de petit degré a été développé (approche ascendante).

Pour l'étape de la construction du classificateur, les modules du choix de critère de poids, d'ajustement de valeurs de seuil ainsi que de la construction de la fonction discriminante ont été conçus.

Finalement, à l'étape de classification de nouvelles observations (tests), les modules d'entrée de données de test et de classification ont été développés. Une description des plus importantes fonctionnalités qui n'ont pas encore été implémentées est incluse dans la section de Conclusions de ce mémoire.

La Figure 3.2 Diagramme d'objets global, montre une vue d'ensemble du logiciel développé suivant la méthodologie UML (Unified Modelling Language), qui est un langage de modélisation d'objets utilisé dans la conception de logiciels.

Il y a 3 grands objets (unités fonctionnelles indépendantes) qui représentent : l'interface graphique du logiciel (LADUI), l'objet responsable de la conversion binaire des données (classe Binarization, objet bi) et l'objet qui spécifie l'algorithme LAD en propre (classe LADAlgorithm, objet o_genPatt).

Pour la classe binarisation, on montre en plus des responsabilités, les différentes méthodes que la classe implémente.

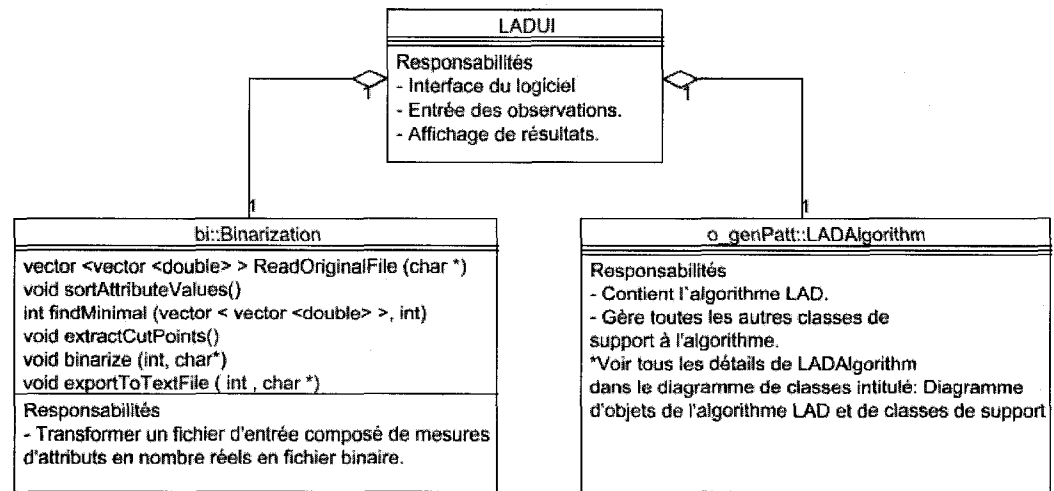


Figure 3.2 Diagramme d'objets global

Le diagramme d'objets suivant montre l'objet o_genPatt de la classe LADAlgorithm et les différentes classes ou modules qui interagissent avec cette dernière afin d'implémenter les différentes étapes de l'algorithme LAD.



Figure 3.3 Diagramme d'objets de l'algorithme LAD et de classes de support

- La classe LADAlgorithm dont l'objet o_genPatt a été créé, est en charge d'implémenter tous les algorithmes de haut niveau tels que la génération de patrons, la lecture de fichiers d'observations de l'historique ainsi que des données de test et l'algorithme de classification de nouvelles observations. La classe LADAlgorithm utilise trois autres classes de support pour ses fonctions, soit la classe PatternAnalysis, la classe singlePattern et la classe foundPatterns.
- La classe PatternAnalysis détermine si un terme généré est un patron prime, un candidat ou un terme qui doit être exclus. Cette classe détermine également si une observation de l'historique est couverte par au moins un patron prime.
- La classe singlePattern est une classe qui sert à la génération et à la manipulation de termes intermédiaires, notamment de T , T' et T'' qui seront définis plus loin lorsque l'algorithme de génération de patrons sera détaillé.
- Finalement la classe foundPatterns représente une collection de patrons primes et de candidats qui ont été trouvés suite à l'application de l'algorithme. Cette classe définit également des opérations sur les patrons, telle que celle d'évaluer la fonction discriminante sur une ou plusieurs observations de test données.

Dans la section suivante, nous allons détailler la méthodologie et les étapes que comporte le LAD de façon plus détaillée.

3.1 La conversion binaire de données.

Si les données d'origine ne sont pas des variables booléennes, l'étape de génération de patrons doit être précédée d'une conversion binaire.

L'algorithme de conversion binaire s'exécute en cinq grandes étapes : la lecture du fichier qui contient les observations d'entraînement, la génération de vecteurs d'indicateurs à ordonnancer, l'ordonnancement des vecteurs d'indicateurs, l'extraction de points de coupure (points d'inflexion) et finalement, la conversion binaire des indicateurs des observations.

La première étape permet d'obtenir une liste d'observations qui sera séparée par indicateurs lors de la deuxième étape. Chaque vecteur d'indicateurs est alors ordonnancé en ordre croissant. L'extraction de points de coupure consiste à repérer les valeurs des indicateurs où l'on passe des observations associées à un état positif vers une valeur dont l'indicateur est associé à un état négatif, et vice-versa. Ces points de coupure obtenus pour chaque vecteur d'indicateurs permettront par simple comparaison de traduire chaque valeur d'indicateur des observations d'entrée en une donnée binaire (1 ou 0).

Alors que la classe Binarisation contient toute l'implémentation de ces étapes, c'est l'interface graphique qui invoque les étapes séquentiellement. À continuation, la Figure 3.4, illustre les grandes étapes telles qu'implémentées dans le logiciel :

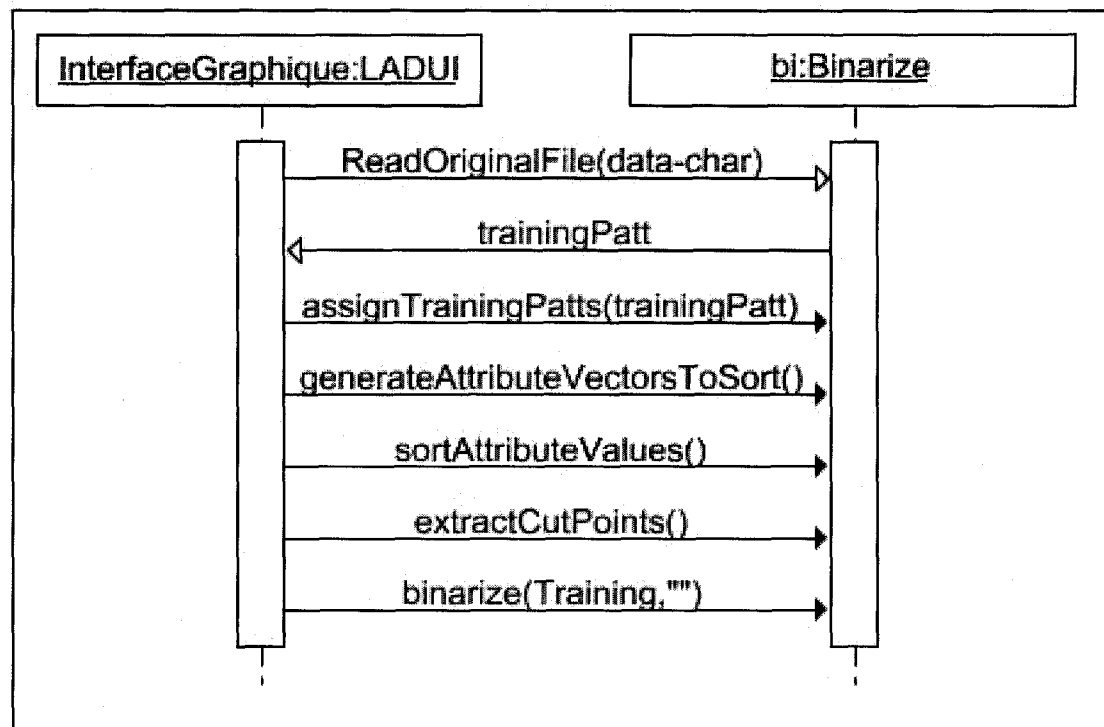


Figure 3.4 Diagramme de séquence des interactions entre la classe interface et la classe binarisation

Dans le diagramme à continuation, Figure 3.5 Diagramme de séquence de la binarisation étape par étape, on montre la séquence d'interactions pour la conversion binaire (dernière étape de la figure précédente). Dans un premier temps, on détermine si la conversion binaire sera effectuée sur des observations d'entraînement ou de test. Après, le fichier d'entrées contenant les observations est lu. Toutes les valeurs des indicateurs de chacune des observations sont comparées avec les points de coupure (points extraits au préalable) définis sur cet indicateur spécifique. Voir Figure 3.6 Diagramme de séquence sur l'analyse de chaque point de coupure, car cette dernière explique en fait les détails d'implémentation auxquels on fait référence dans la dernière étape (réf. [Analyze

each cutpoint]) de la Figure 3.5 Diagramme de séquence de la binarisation étape par étape.

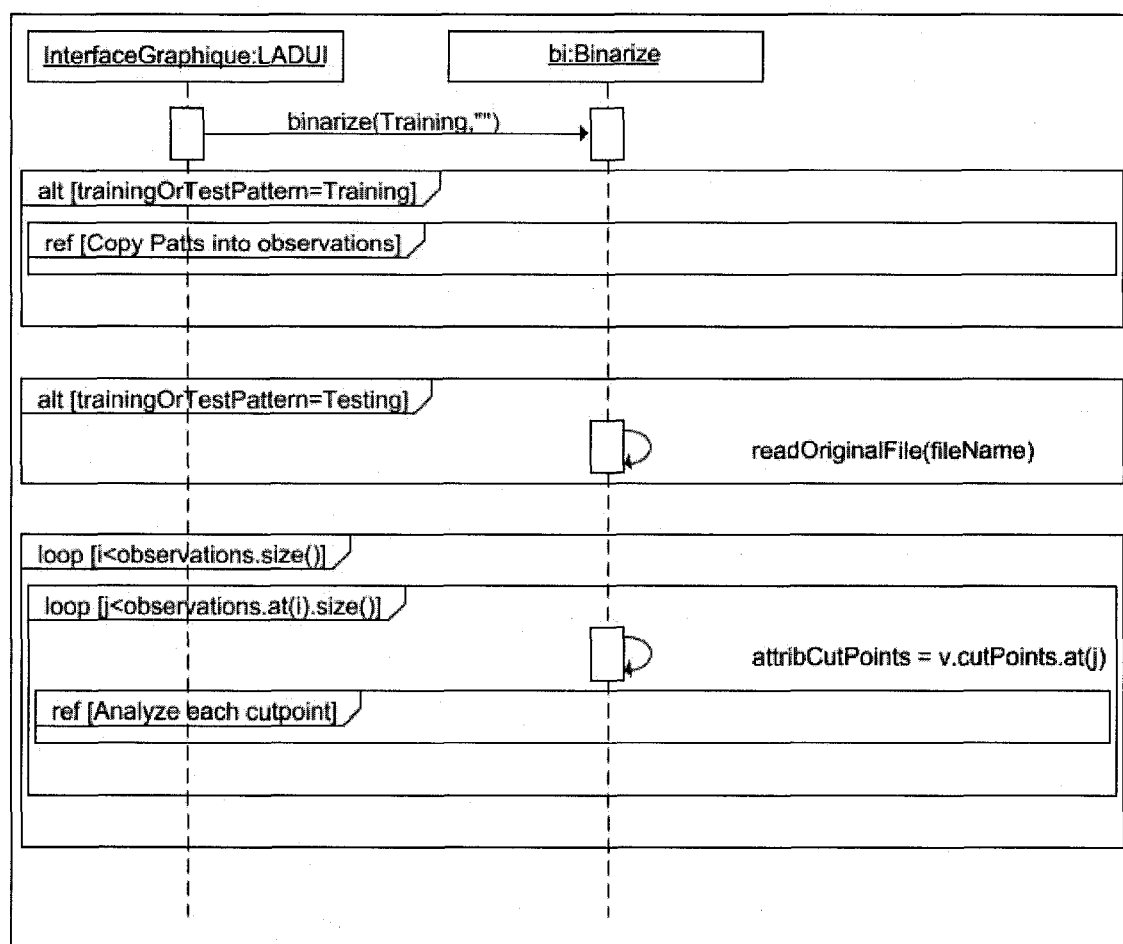


Figure 3.5 Diagramme de séquence de la binarisation étape par étape

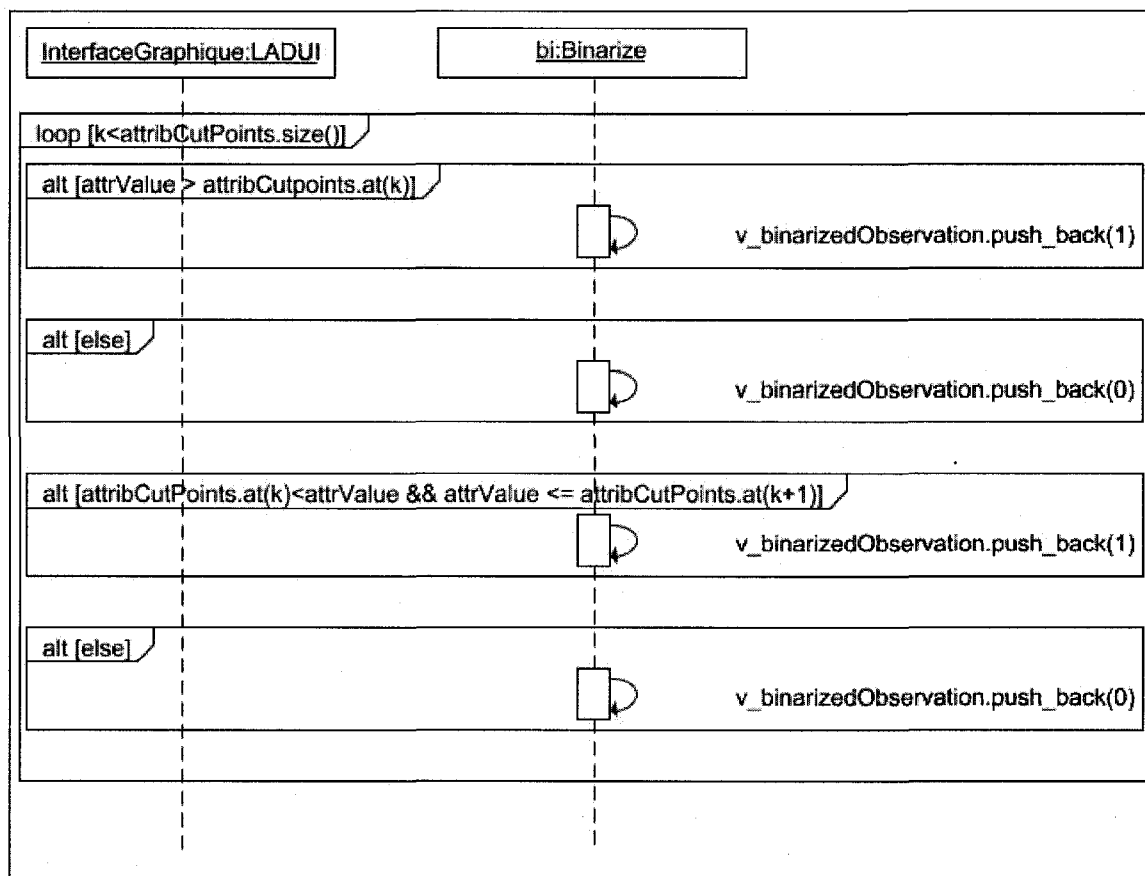


Figure 3.6 Diagramme de séquence sur l'analyse de chaque point de coupure

Dans le diagramme précédent, on s'aperçoit que pour chaque point de coupure, deux comparaisons sont faites. Pour les variables de niveau, (voir section 2.1.1 La conversion binaire de données), la condition évaluée est `attrValue > attribCutpoints.at(k)` alors que pour les variables d'intervalles, la condition évaluée est `attribCutPoints.at(k) < attrValue && attrValue <= attribCutPoints.at(k+1)`. Dépendamment du résultat de ces comparaisons, on assignera la valeur 0 ou 1 à l'état (`v_binarizedObservation.push_back(0)`, `v_binarizedObservation.push_back(1)`).

3.2 Implémentation de l'algorithme de génération de patrons

Ici, nous traiterons l'algorithme de génération de patrons primes positifs, le cas de patrons primes négatifs étant l'analogue. Les diagrammes de séquence présentés à continuation à la Figure 3.7 Génération de patrons 1ère partie, à la Figure 3.8 Génération de patrons, 2ème partie, à la Figure 3.9 *findIfInCandidate*, référence de support à la génération de patrons, et à la Figure 3.10 *AnalyzeIfCandidateOrPrime*, référence de support à la génération de patrons, aideront à la compréhension des interactions entre les objets et les messages échangés entre ceux-ci. Voici la description des principales étapes de l'algorithme de génération de patrons *findPatterns* de la classe *LADAlgorithm*.

Dans un premier temps, dans la Figure 3.7 Génération de patrons 1ère partie, l'algorithme génère les termes de degré 1 qui couvrent au moins une observation positive. S'il s'agit des patrons primes, on les ajoute à la liste de patrons primes de l'objet *primCand* de la classe *foundPatterns*, sinon, s'il s'agit de termes candidats, on les ajoute à la liste de candidats du même objet *primCand*.

Par la suite, pour chacun de candidats contenus dans la liste de l'objet *primCand*, on aura cette même boucle, ici illustrée pour le degré d :

On assigne le candidat de degré d à l'objet T de la classe *singlePattern* et on trouve le littéral qui a le plus grand ordre lexicographique contenu dans T que l'on nomme p . On joint le littéral dont l'ordre lexicographique est $p+1$ à T , et on assigne ce nouveau terme à T_{prim} , que devient un terme de degré $d+1$.

Pour que T soit conservé comme un patron prime ou un candidat, il faut que tous les sous-termes de T_{prim} de degré d soient aussi des candidats de degré $d-1$. Dans l'algorithme, on assigne à $T2_{\text{prim}}$ le terme obtenu d'enlever un littéral à la fois de T_{prim} . Si tous les sous-termes de T_{prim} sont des candidats de degré $d-1$, on a deux variantes : si T_{prim} est un patron prime, alors on l'ajoute à la liste de patrons primes, autrement on l'ajoute à la liste de candidats de degré d pour l'itération suivante (étude de termes de degré $d+1$). Dans le restant des cas, on ne considère pas T_{prim} pour les itérations ultérieures.

Au cycle suivant, la démarche se répète avec la liste de candidats de degré $d+1$ qui a été générée lors de l'étape précédente.

Les quatre diagrammes présentés à continuation correspondent à la méthode `findPatterns` qui a été implémentée dans la classe `LADAlgorithm`, et qui implémente la génération de patrons. Cette méthode est illustrée ici dans quatre diagrammes différents (Figure 3.7, Figure 3.8, Figure 3.9, Figure 3.10 et Figure 3.11) afin d'assurer la clarté de la séquence des interactions.

Le premier diagramme, Figure 3.7 Génération de patrons 1ère partie, illustre la génération et l'analyse de termes de degré 1, alors que la Figure 3.8 Génération de patrons, 2ème partie illustre la génération et l'analyse de termes de degré supérieur. Il est nécessaire de mentionner quelques conventions : lorsqu'un contour délimite un groupe d'interactions, ceci veut dire qu'il s'agit d'une boucle, d'une alternative ou d'une référence qui envoie sur un autre diagramme.

En particulier, le groupe *loop* veut dire que tant et aussi longtemps que la condition énoncée est valide, la séquence d'interactions dans les limites du contour sera exécutée de façon répétée. Par exemple, l'entête *loop* [$d \leq upToD$] veut dire que tant et aussi longtemps que d sera plus petit que $upToD$, le code à l'intérieur du contour sera exécuté.

De façon analogue, si on trouve l'étiquette *alt*, ceci représente un groupement de séquences dont l'exécution peut ou ne peut pas s'exécuter dépendamment si la condition énoncée s'avère vraie. C'est ainsi que l'étiquette *alt* [$isPrime = true$], veut dire que la suite de séquences dans le rectangle s'exécutera si et seulement si la valeur retournée à l'étape précédente *isPrime* est vraie.

Si on trouve l'étiquette *ref*, on doit se référer à un diagramme connexe car par souci de clarté et dans le but d'assurer la compréhension du fonctionnement global de la séquence, certains détails ont été traités séparément. C'est le cas de *ref*[*findIfInCandidate*] et de *ref*[*AnalyzeIfCandidateOrPrime*] qui apparaissent dans la Figure 3.8 et qui se réfèrent à la Figure 3.9, Figure 3.10.

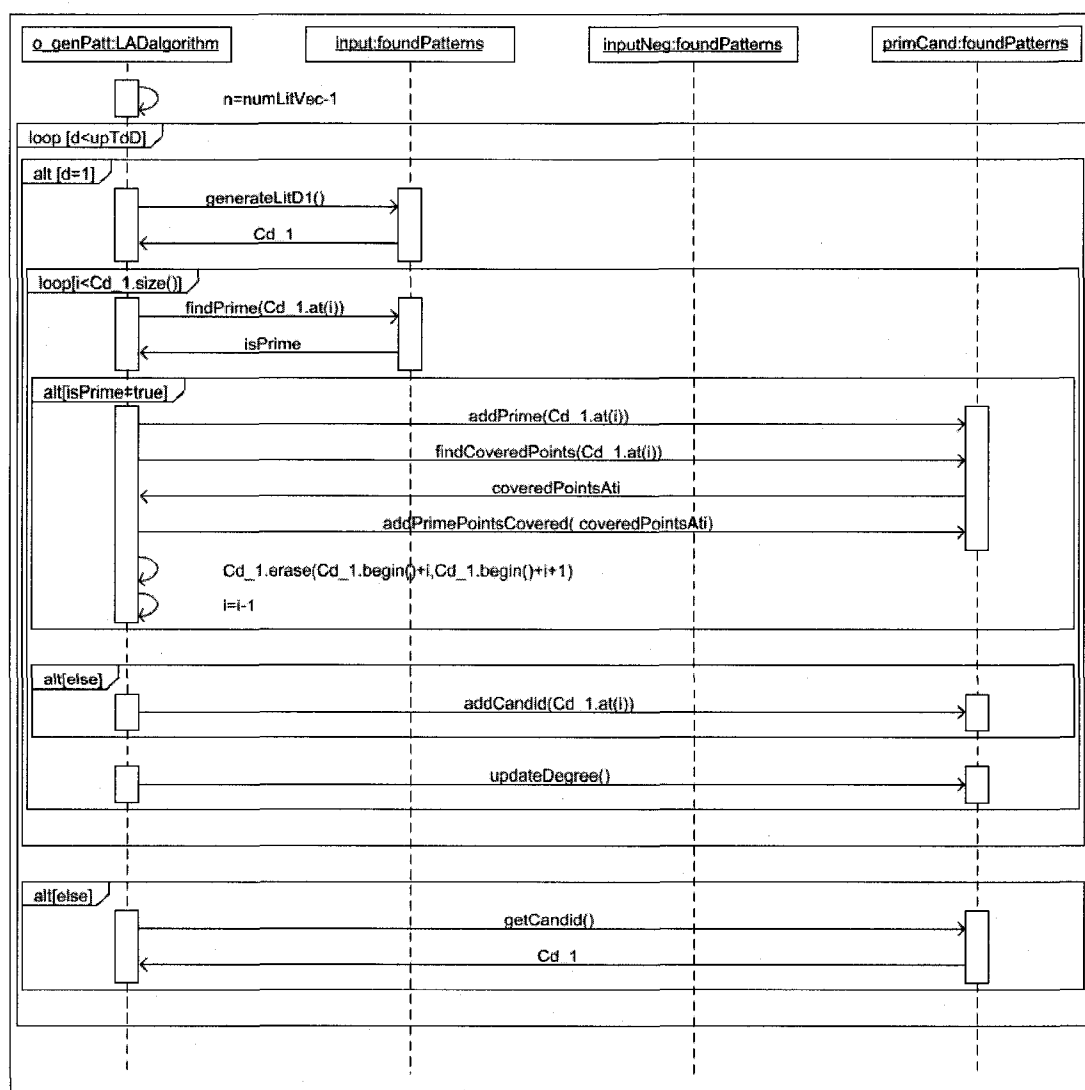
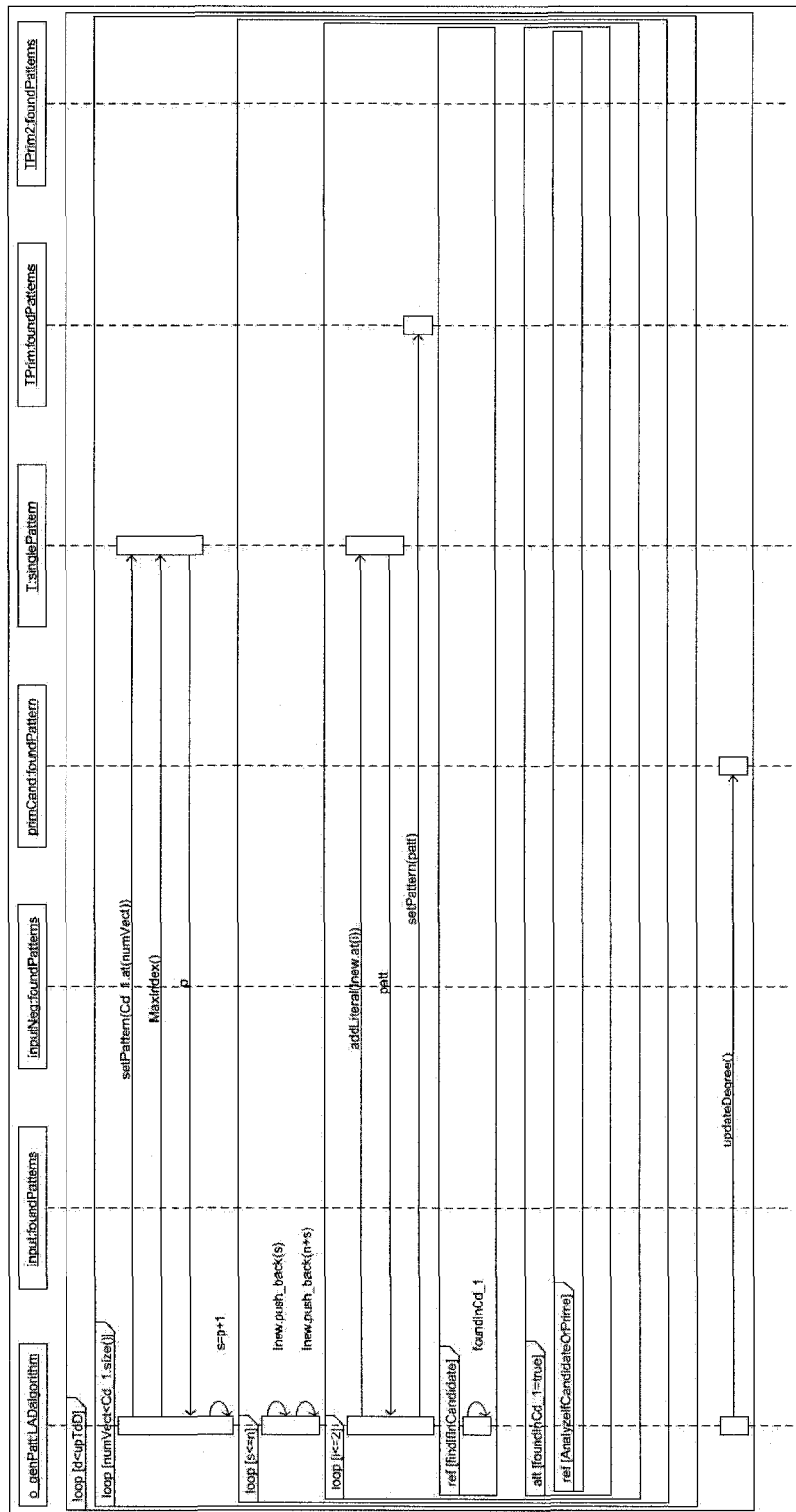


Figure 3.7 Génération de patrons 1^{ère} partie

Figure 3.8 Génération de patrons, 2^{ème} partie

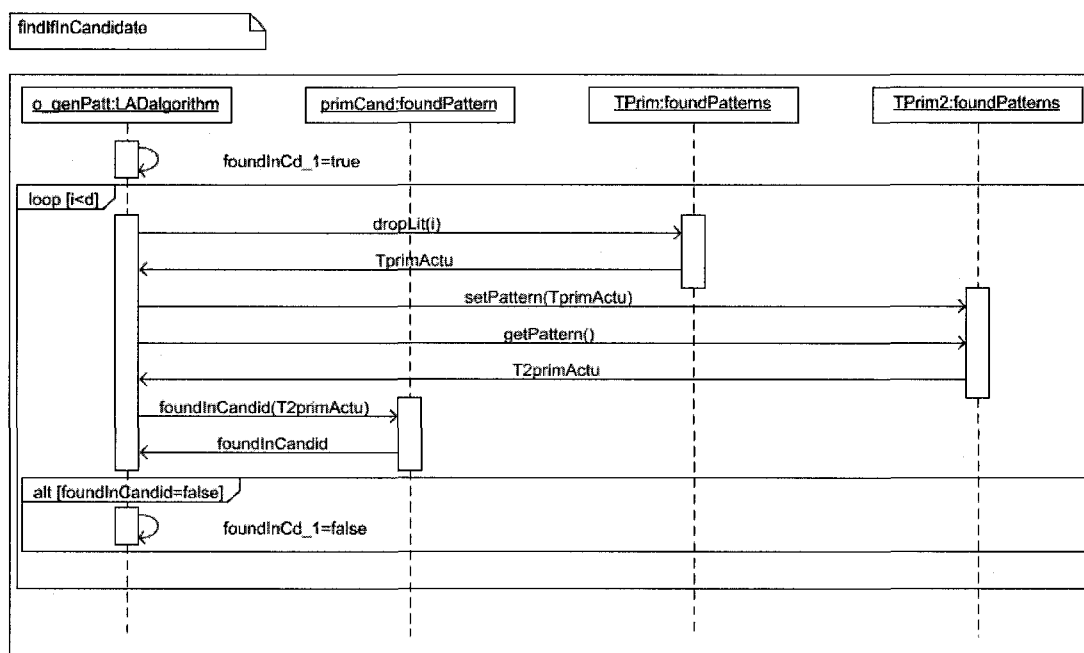


Figure 3.9 `findIfInCandidate`, référence de support à la génération de patrons

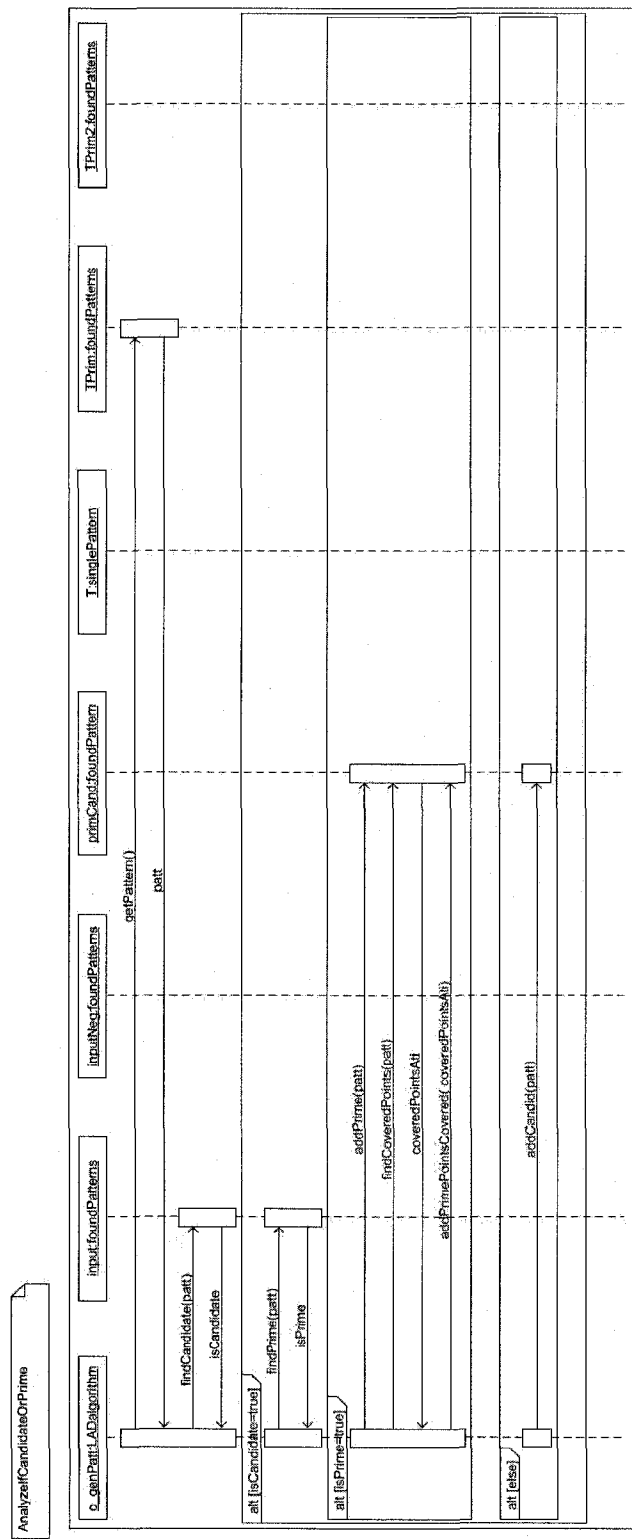


Figure 3.10 `AnalyzeIfCandidateOrPrime`, référence de support à la génération de patrons

3.3 Formation de la fonction discriminante et classification

Dans le logiciel, lorsque l'utilisateur indique qu'il désire fournir un ensemble de tests les étapes suivantes s'exécutent en ordre séquentiel : lecture du fichier de tests (contenant des observations converties en format binaire au préalable), évaluation des observations par les patrons, obtention de poids de patrons et évaluation de la fonction discriminante.

D'un point de vue des interactions, voici un aperçu détaillé des interactions illustrées dans la Figure 3.11 Formation de la fonction discriminante et classification de nouvelles observations. :

- L'objet *o_genPatt* de la classe *LADAlgorithm* fait appel à la méthode de lecture de fichiers de texte *orderReadFile* de l'objet *input* de la classe *foundPatterns*. Le vecteur *testResp* contenant une représentation du fichier de réponses est retourné.
- L'objet *o_genPatt* fait appel à la fonction *evalDiscriminantFunction* de l'objet *primCand* de la classe *foundPatterns* afin d'évaluer les observations de tests par les patrons positifs. Le vecteur *posTruthVal* qui contient la liste de patrons validés pour chaque observation de l'ensemble de tests est retourné.
- L'objet *o_genPatt* fait appel à la fonction *evalDiscriminantFunction* de l'objet *negPrimCand* de la classe *foundPatterns* afin d'évaluer les observations de tests par les patrons négatifs. Le vecteur *negTruthVal* qui contient la liste de patrons validés pour chaque observation de l'ensemble de tests est retourné.

- L'objet *o_genPatt* fait appel à la fonction *getNumPointsCovered* de l'objet *primCand* , pour obtenir la liste d'observations de l'historique couverte par chaque patron positif. Le vecteur *ptCov* est retourné.
- L'objet *o_genPatt* fait appel à la fonction *getNumPointsCovered* de l'objet *negPrimCand* pour obtenir la liste d'observations de l'historique couverte par chaque patron négatif. Le vecteur *negPtCov* est retourné.
- Exécution d'une boucle (`loop[i<posTruthVal.size()]`) de calcul de la fonction discriminante en ce qui a trait aux patrons positifs pour chaque observation (*posSum*): Obtention de poids de l'*i*^{ème} patron positif (`ptCov.at(i).size()`), obtention de valeur booléenne qui indique si *i*^{ème} patron positif valide l'observation en question.
- Exécution d'une boucle de calcul (`loop[i<negTruthVal.size()]`) de la fonction discriminante en ce qui a trait aux patrons négatifs pour chaque observation (*negSum*).
- Pour chaque observation, on soustrait le score de la fonction discriminante liée aux patrons négatifs du score de la fonction discriminante liée aux patrons positifs. (`sumPosVect.at(i) – sumNegVect.at(i)`)
- Le vecteur *_ret_testResp* contient le score de la fonction discriminante pour chaque observation de tests.

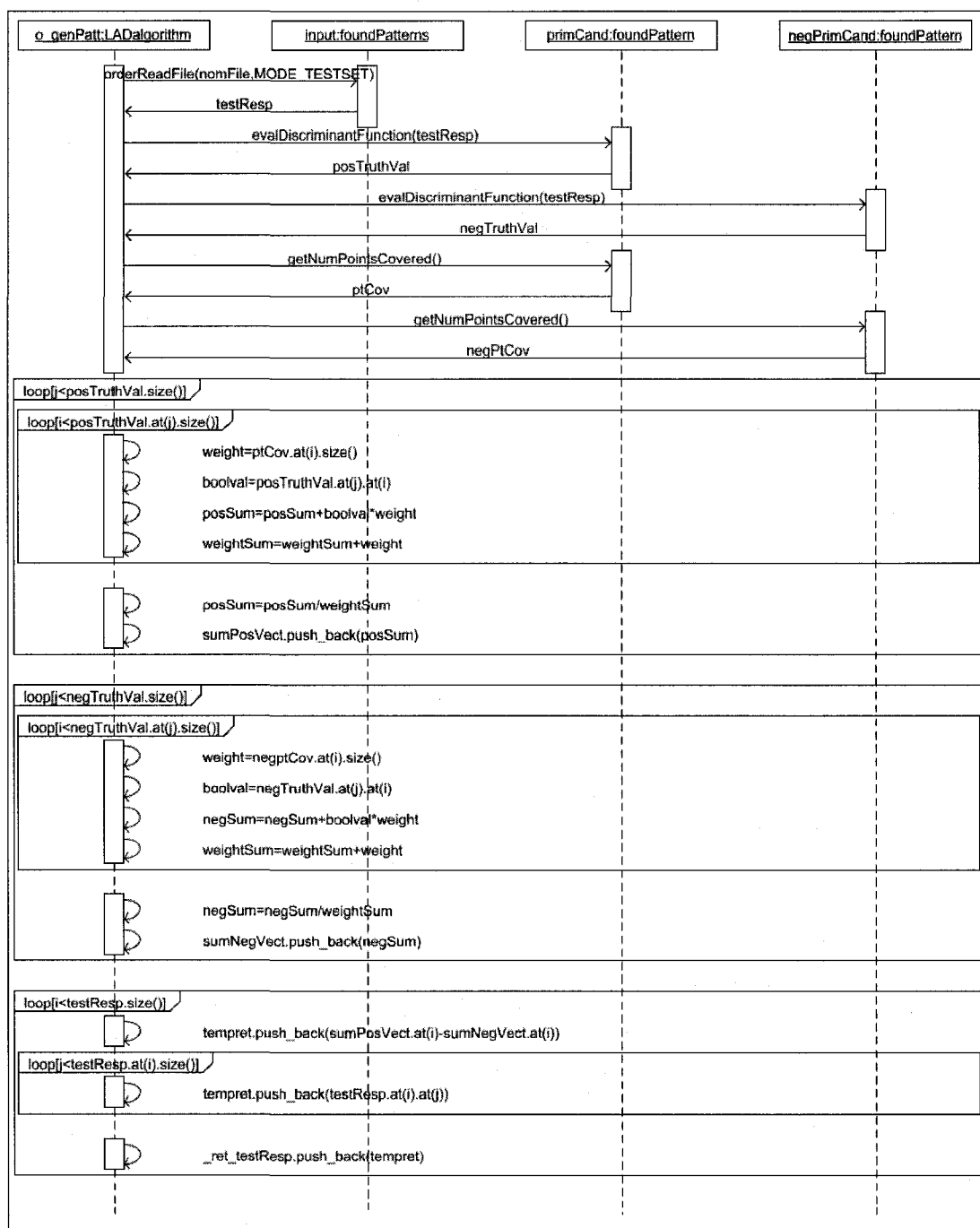


Figure 3.11 Formation de la fonction discriminante et classification de nouvelles observations.

CHAPITRE 4

APPLICATIONS À LA MAINTENANCE CONDITIONNELLE

Nous allons étudier l'utilisation du logiciel LAD-CBM dans le cadre de deux applications de maintenance de machines. Nous avons cherché des données industrielles de maintenance publiées dans la littérature de maintenance, ayant comme critères d'avoir des observations qui ont déjà été nettoyées et qui n'avaient pas de valeurs manquantes. Il était également important d'avoir un ensemble de tests pour lequel on connaissait l'état à priori (même si cette information n'était pas utilisée dans le logiciel), à des fins d'évaluation de la performance du logiciel LAD-CBM.

Nous allons donc décrire les résultats des étapes du LAD appliqués à deux études de cas. Dans un premier temps, nous étudierons le cas d'un mécanisme de roulement pour lequel on a 8 observations et 6 indicateurs réels. Dans un deuxième temps, nous traiterons le cas d'un transformateur de courant pour lequel on a 50 observations et 5 indicateurs réels.

Vu que les observations se constituent des indicateurs réels, et tel qu'il a été décrit en 2.1.1 La conversion binaire de données, nous ne pourrons pas interpréter les patrons directement dans le domaine du problème. Une solution est proposée dans la section Conclusions et Futures Recherches.

Toutes les expériences du logiciel LAD-CBM ont été conduites sur un ordinateur muni d'un processeur de 32 bit Intel Pentium-M 760 de 2.13 GHz avec 2 Go de mémoire vive sur Windows Vista.

4.1 Maintenance conditionnelle d'un mécanisme de roulement

Dans l'étude de (Sun Q., et al., 2004), des capteurs ont été installés près d'un mécanisme de roulement d'un appareil de forage et les données de vibration obtenues ont été analysées dans le domaine temporel.

À titre indicatif, la Figure 4.1, montre un mécanisme de roulement générique ainsi que les composants où des défauts arrivent le plus fréquemment :

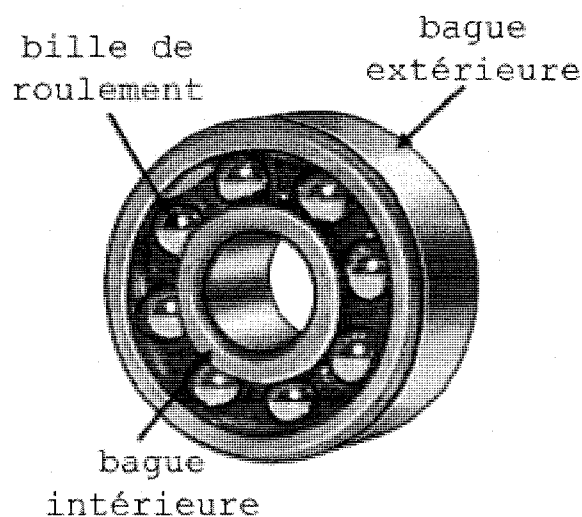


Figure 4.1 Diagramme d'un roulement mécanique à billes

Pour chaque signal capturé à 8 intervalles différents, six indicateurs de l'amplitude de vibration ont été calculés, il s'agit de données dans le domaine du temps. Ces paramètres sont : facteur de crête (Cf, par son sigle en anglais), la valeur de Kurtosis (Kv), le facteur de marge (CI), le facteur d'impulsion (If), la racine carrée relative (Rv) y la valeur relative entre pics (Pk). Ces données sont présentées dans le Tableau 4.1.

Tableau 4.1 Observations de maintenance conditionnelle d'un mécanisme de roulement (Sun Q., et al., 2004)

État	# Mesure	Rv	Pk	Cf	Kv	CI	If
Défectueux	1	7.8491	22.987	2.9301	3.4406	6.3869	3.7442
	2	4.9825	12.5721	2.5233	3.1313	5.6337	3.2916
	3	6.0294	15.125	2.5085	2.5077	5.3118	3.1450
	4	10.3168	31.1830	3.0225	3.5191	6.9424	3.9892
Normal	5	1.2348	3.8428	3.1129	3.6318	7.814	4.2400
	6	0.8953	2.6677	2.9707	3.2602	6.7993	3.8483
	7	0.9193	2.8486	3.0987	3.4304	7.2580	4.0608
	8	1.069	3.5156	3.2887	3.4158	8.5502	4.5622

La première étape consiste à définir un ensemble d'apprentissage et un ensemble de tests. Dans ce cas-ci, les données étant peu nombreuses, on a formé de façon aléatoire des groupes de 6 observations qui seraient l'ensemble d'apprentissage et les 2 observations restantes, une positive et l'autre négative serviraient comme des observations de tests.

Il existe $\binom{4}{1} \times \binom{4}{1} = 4 \times 4 = 16$ (c'est-à-dire le nombre de façons de choisir une observation positive parmi les quatre observations positives existantes multiplié par le nombre de façons de choisir une observation négative parmi les quatre observations

négatives existantes) manières différentes de choisir ces ensembles et nous avons étudié toutes les combinaisons possibles en sachant que tout ensemble de tests devra être composé d'une observation négative et d'une observation positive.

Tableau 4.2 Plan d'expériences pour l'apprentissage et le test du mécanisme de roulement

Essai	Ensemble d'apprentissage (observations)	Ensemble de tests (observations)
1	2,3,4,6,7,8	1,5
2	2,3,4,5,7,8	1,6
3	2,3,4,5,6,8	1,7
4	2,3,4,5,6,7	1,8
5	1,3,4,6,7,8	2,5
6	1,3,4,5,7,8	2,6
7	1,3,4,5,6,8	2,7
8	1,3,4,5,6,7	2,8
9	1,2,4,6,7,8	3,5
10	1,2,4,5,7,8	3,6
11	1,2,4,5,6,8	3,7
12	1,2,4,5,6,7	3,8
13	1,2,3,6,7,8	4,5
14	1,2,3,5,7,8	4,6
15	1,2,3,5,6,8	4,7
16	1,2,3,5,6,7	4,8

Les observations résultant de la conversion au format binaire pour l'essai 1 sont présentées dans le Tableau 4.3.

Tableau 4.3 Observations en format binaire pour le premier essai - Mécanisme de roulement

Essai 1- Ensemble d'apprentissage	Essai 1 - Ensemble de tests
1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	0,1,1,1,1,0,0,0,1,1,0,1,1,0,0,0,1,1,0,0,0
1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	0,0,0,1,0,1,0,1,1,0,1,1,0,1,0,1,1,0,1,0,1
1,1,1,1,0,1,1,0,1,0,1,1,0,1,1,0,1,0,1,1,0	
0,0,0,1,1,0,0,0,1,1,0,1,1,0,0,0,1,1,0,0,0	
0,0,0,1,0,1,0,1,1,1,0,1,0,1,0,1,1,0,1,0,1	
0,0,0,1,0,1,0,1,1,1,0,1,0,1,0,1,1,0,1,0,1	

Tableau 4.4 Synthèse de la conversion binaire des observations du mécanisme de roulement

# Essai	# Indicateurs binaires
Essai 1	20
Essai 2	10
Essai 3	22
Essai 4	22
Essai 5	20
Essai 6	10
Essai 7	22
Essai 8	22
Essai 9	20
Essai 10	10

Tableau 4.4 Synthèse de la conversion binaire des observations du mécanisme de roulement (suite et fin)

# Essai	# Indicateurs binaires
Essai 11	22
Essai 12	22
Essai 13	8
Essai 14	10
Essai 15	10
Essai 16	10

Selon la convention adoptée, l'état normal correspond à 0, alors qu'on assigne l'état 1 aux états de défaut.

Avec les observations en format binaire, nous procédons à la génération de patrons. Lors du premier essai, 49 patrons négatifs et 38 patrons positifs ont été trouvés. Voici les cinq premiers patrons négatifs et les cinq premiers patrons positifs obtenus lors de l'essai 1, à partir de l'ensemble d'apprentissage en format binaire. Le restant de patrons de chaque essai se trouve à l'Annexe B. Les patrons positifs sont ceux qui qualifient l'état comme défectueux alors que les patrons négatifs le qualifient comme normal. Dans cette liste de patrons, on adopte la notation du logiciel en anglais et on suit le format suivant : (Patron positif ou négatif) (#identifiant du patron) : (Indicateurs booléens qui forment le patron) (Identifiant des observations couvertes par le patron). De plus, lorsqu'un indicateur est précédé par NOT, ceci indique le caractère négatif au sens booléen de l'indicateur.

Tableau 4.5 Cinq patrons positifs et cinq premiers patrons négatifs du premier essai – Mécanisme de roulement

Identifiant du patron	Indicateurs du patron	Observations couvertes
Negative pattern #1:	NOT 1	Observations couvertes par ce patron: 4,5,6
Negative pattern #2	NOT 2	Observations couvertes par ce patron: 4,5,6
Negative pattern #3:	4	Observations couvertes par ce patron: 4
Negative pattern #4	7	Observations couvertes par ce patron: 5,6
Negative pattern #5:	9	Observations couvertes par ce patron: 4,5,6
Positive pattern #1:	1	Observations couvertes par ce patron: 1,2,3
Positive pattern #2:	2	Observations couvertes par ce patron: 1,2,3
Positive pattern #3:	NOT 3	Observations couvertes par ce patron: 1,2
Positive pattern #4:	6	Observations couvertes par ce patron: 3
Positive pattern #5:	NOT 8	Observations couvertes par ce patron: 1,2

Dans cet exemple, on remarque que certaines observations sont couvertes par plus d'un patron. Un patron a est redondant, s'il existe un autre patron b qui couvre toutes les observations couvertes par le patron a , en plus de couvrir d'autres observations.

Bien que dans la phase de génération de patrons, on vise à capturer tous les aspects significatifs d'un phénomène, il est souvent inefficace d'utiliser un nombre trop élevé de patrons. D'une part, ceci rend plus difficile l'interprétation des résultats de classification car il faudrait analyser un plus grand nombre de résultats positifs de chaque patron individuel, et d'autre part, l'utilisation de patrons redondants ne contribuerait que marginalement au résultat final.

Voici une synthèse du nombre de patrons générés dans chacun des essais et le temps d'exécution pour les trouver.

Tableau 4.6 Synthèse de la génération de patrons - Mécanisme de roulement

Essai	Nombre d'indicateurs binaires	Nombre de patrons générés	Temps d'exécution
1	20	49 patrons négatifs 38 patrons positifs	5 secondes 260 ms
2	10	10 patrons négatifs 10 patrons positifs	2 secondes 200 ms
3	22	61 patrons négatifs 49 patrons positifs	5 secondes 600ms
4	22	61 patrons négatifs 49 patrons positifs	4 secondes 300 ms
5	20	43 patrons négatifs 38 patrons positifs	2 s 200ms
6	10	10 patrons négatifs 10 patrons positifs	4s 200ms

Tableau 4.6 Synthèse de la génération de patrons - Mécanisme de roulement (suite et fin)

Essai	Nombre d'indicateurs binaires	Nombre de patrons générés	Temps d'exécution
7	22	52 patrons négatifs 52 patrons positifs	5s 500ms
8	22	52 patrons négatifs 52 patrons positifs	5s 800ms
9	20	43 patrons négatifs 38 patrons positifs	2s 500ms
10	10	10 patrons négatifs 10 patrons positifs	1s 100ms
11	22	52 patrons négatifs 52 patrons positifs	4s 500ms
12	22	52 patrons négatifs 52 patrons positifs	4s 600ms
13	8	7 patrons négatifs 8 patrons positifs	500ms
14	10	10 patrons négatifs 10 patrons positifs	1s 300ms
15	10	10 patrons négatifs 10 patrons positifs	1s 200ms
16	10	10 patrons négatifs 10 patrons positifs	1s 100ms

Dans l'application précédente, le nombre de patrons (voir Annexe B pour la liste complète de patrons) est grand par rapport au petit nombre d'observations dans l'ensemble d'apprentissage. Il serait possible d'éliminer certains patrons qui couvrent des cas rares et spécifiques qui sont déjà couverts par de patrons plus généraux, sans que

les résultats ne se dégradent. De cette façon, on arriverait à avoir un ensemble de patrons minimal qui décrit bien le phénomène en question. Cet ensemble minimal est appelé ensemble de couverture. La génération d'un tel ensemble dépasse le but de ce travail, mais une description plus en détails est incluse dans la section Conclusions.

Il est aisé de constater que malgré le fait que la génération de patrons a été conduite afin de détecter les patrons jusqu'au degré 5 et que les observations ont été binarisées en 11 ou plus d'indicateurs binaires (selon l'essai spécifique), tous les patrons sont de maximum degré 2. Ceci veut dire que les patrons ont un plus grand pouvoir généralisateur (comme ils sont plus généraux, ils couvrent un plus grand nombre des observations de l'ensemble d'apprentissage) et en même temps qu'un petit nombre d'indicateurs binaires, sont nécessaires pour qualifier une observation.

Voici les résultats de classification de chacun des essais. Dans chacun des essais, dans les données de tests, la première observation correspond à une observation positive (état défectueux) alors que la deuxième correspond à l'état normal. L'état était connu à priori, afin de pouvoir calculer l'efficacité de classification de l'algorithme.

Tableau 4.7 Résultats numériques de classification - Mécanisme de roulement

Essai	Résultats de classification
1	-0.567059
	-0.593726
2	0.333333
	0.416667

Tableau 4.7 Résultats numériques de classification - Mécanisme de roulement (suite)

Essai	Résultats de classification
3	-0.561432
	-0.216346
4	-0.561432
	-0.82906
5	0.693333
	-0.566667
6	0.333333
	-0.166667
7	0.490196
	0.196078
8	0.490196
	-0.862745
9	0.813333
	-0.566667
10	0.833333
	0.333333
11	0.754902
	0.196078
12	0.754902
	-0.862745
13	0.047619
	-0.52381
14	0.0833333

Tableau 4.7 Résultats numériques de classification - Mécanisme de roulement (suite et fin)

Essai	Résultats de classification
	0.416667
15	0.0833333
	-0.416667
16	0.0833333
	-0.916667

Pour définir si l'état d'une observation devrait être classifié comme défectueux ou normal à partir de la valeur de classification, on doit définir de valeurs seuil. Ces valeurs sont dépendantes du problème. Une des approches utilisées serait celle de minimiser l'erreur de faux négatifs et/ou de faux positifs. Cependant, avec un si petit nombre de données historiques, cette procédure n'est pas utile car l'influence de valeurs aberrantes peut mettre en compromis la validité de la valeur seuil choisie.

Généralement, l'approche serait de définir des valeurs seuil de classification et de classifier des nouvelles observations. Une fois l'état réel de ces observations connu, on calculerait les mesures d'efficacité (faux positif, faux négatif et décisions correctes) de classification et on ajusterait les valeurs seuil au besoin afin d'améliorer les performances.

Dans notre cas, on n'a pas un grand banc de données de maintenance qui nous permettrait de simuler l'arrivée de nouvelles observations. Empiriquement, nous pouvons par contre, évaluer les données d'apprentissage par la fonction de classification

afin de voir quels sont les scores minimaux et maximaux de classification obtenus tant pour les observations positives que négatives.

Nous avons privilégié cette approche et ensuite créé une zone d'indécision autour du point neutre (généralement 0). Toute tentative d'élargir les zones positives et/ou négatives doit se faire de manière exploratoire (la valeur spécifique doit être trouvée par essai et erreur) ayant comme critère de réduire la zone d'indécision sans pour autant augmenter le risque de défaillance non-détectée.

Dans nos essais sur les observations de maintenance de cette application de mécanisme de roulement, ainsi que sur les observations du transformateur de courant traité dans la section 4.2 Maintenance conditionnelle d'un transformateur de courant, nous avons recensé que généralement, le fait d'établir une zone d'indécision de longueur 0.1 (5% de la longueur de l'échelle de classification -1 à +1) et centrée à -0.1 du point neutre, est une bonne pratique. Ceci représente un bon compromis entre la prédiction correcte des observations positives et les faux positifs et constitue un moyen adéquat pour diminuer l'influence de la zone d'indécision sur l'exactitude de la fonction de classification.

Le choix de centrer la zone d'indécision à -0.1 du point neutre, provient du fait qu'en maintenance, on préfère détecter un risque de défaillance (associé à l'état positif des observations) que de le laisser passer inaperçu, quitte à générer quelques fausses alarmes. En déplaçant la zone d'indécision à gauche du 0, on élargit la zone de classification positive.

Dans le Tableau 4.8, nous présentons un exemple de calcul.

Tableau 4.8 Exemple de calcul de la zone d'indécision et des zones de décision - Mécanisme de roulement

Essai 1	
Observations positives Val Max = 0.826667 ; Val Min= 0.65333 ;	
Observations négatives Val Max = -0.705882; Val Min = -0.852941;	
Point neutre = Val Max (Obs. négatives) + Val min (Obs. positives) = -0.052	
Zone d'indécision	
Centrée sur : Point neutre - 0.1 = - 0.052 - 0.1 = -0.152	
Borne inférieur -0.152-0.05 = -0.20	
Borne supérieur -0.152+0.05 = -0.10	
Décision de classification	
État = Positif si	Résultat \geq -0.10
État = Négatif si	Résultat \leq -0.20
Zone d'indécision	-0.20 < Résultat < -0.10

Tableau 4.9 Résultats finaux de la classification - Mécanisme de roulement

# Essai	Résultat	Décision de classification	État déduit	État réel
1	-0.567059	Positif si \geq -0.10	Négatif	Positif
	-0.593726	Négatif si \leq -0.20	NÉGATIF	NÉGATIF
2	0.333333	Positif si \geq -0.05	POSITIF	POSITIF
	0.416667	Négatif si \leq -0.15	Positif	Négatif
3	-0.561432	Positif si \geq -0.08	Négatif	Positif
	-0.216346	Négatif si \leq -0.18	NÉGATIF	NÉGATIF
4	-0.561432	Positif si \geq -0.08	Négatif	Positif
	-0.82906	Négatif si \leq -0.18	NÉGATIF	NÉGATIF

Tableau 4.9 Résultats finaux de la classification - Mécanisme de roulement (suite)

# Essai	Résultat	Décision de classification	État déduit	État réel
5	0.693333	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.566667	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
6	0.333333	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.166667	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
7	0.490196	Positif si ≥ -0.05	POSITIF	POSITIF
	0.196078	Négatif si ≤ -0.15	Positif	Négatif
8	0.490196	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.862745	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
9	0.813333	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.566667	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
10	0.833333	Positif si ≥ -0.05	POSITIF	POSITIF
	0.333333	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
11	0.754902	Positif si ≥ -0.05	POSITIF	POSITIF
	0.196078	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
12	0.754902	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.862745	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF
13	0.047619	Positif si ≥ -0.15	POSITIF	POSITIF
	-0.52381	Négatif si ≤ -0.25	NÉGATIF	NÉGATIF
14	0.0833333,	Positif si ≥ -0.05	POSITIF	POSITIF
	0.416667	Négatif si ≤ -0.15	POSITIF	NÉGATIF
15	0.0833333	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.416667	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF

Tableau 4.9 Résultats finaux de la classification - Mécanisme de roulement (suite et fin)

# Essai	Résultat	Décision de classification	État déduit	État réel
16	0.0833333	Positif si ≥ -0.05	POSITIF	POSITIF
	-0.916667	Négatif si ≤ -0.15	NÉGATIF	NÉGATIF

Au cours de ces essais, le taux d'efficacité de précision déduit du Tableau 4.9 est de 84.37% (27 classifications correctes sur 32 observations), le taux de faux négatifs est de 9.375% (3 observations sur 32 observations classifiées comme négatives, alors que l'état réel est positif) et le taux de fausses alarmes est de 6.25% (2 observations sur 32 observations classifiées comme positives, alors que l'état réel est négatif).

Bien que cette performance ne soit pas tout à fait satisfaisante, il faut tenir compte du fait que le nombre restreint d'observations disponibles pour l'apprentissage (4 observations) est une limitation à l'étape d'apprentissage.

Tel qu'il arrive avec les approches statistiques qui ont besoin d'un bon nombre d'observations pour qu'une analyse ne soit pas faussée par des données aberrantes, la performance du LAD dépend également du volume de données. Dans l'exemple suivant, nous allons traiter une application avec un plus grand nombre d'observations.

4.2 Maintenance conditionnelle d'un transformateur de courant

La deuxième application est reliée aux données analytiques de gaz dissout (DGA, Dissolved gas analysis) comme méthode diagnostique de détection de défaillances internes des transformateurs de courant. Ces gaz sont produits par la dégradation de

l'huile de transformateur et de composants et d'isolants contenus dans le transformateur.

Cette dégradation est accélérée en présence d'un défaut structurel du transformateur.

Dans (Lv G., et al. 2004), on présente des données expérimentales de DGA. Quatre états existent : État normal, réchauffement thermique, décharge à basse énergie et décharge à haute énergie.

La base du diagnostic, ce sont les caractéristiques extraites de données brutes de cinq indicateurs qui sont présentées dans le Tableau 4.10:

Tableau 4.10 Observations de l'historique - Transformateur de courant

État	Observations d'apprentissage (valeurs réelles)
Décharge de haute énergie	40.8,3,3.6,3.5,7.4 23,12,12,10,61 57,13,0.1,11,12 31,6.6,19,4.7,67 285,36.3,75,8.4,133 5283,179,320,3020,2314 335,67,18,143,170 44,12.2,3.4,3.2,17.4 42,62,5,63,73 60,40,9.9,110,70 240,28,6,26,85 466.5,148.8,13,266,511 260,130,29,84,92 250,63,3.8,66,120 59,7.1,19,4.5,71
Décharge de basse énergie	650,53,34,20,0 0,5.2,5.12,9.58,14.6 1565,93,34,47,0 160,90,27,17,5.8 35,25,0,23,22

Tableau 4.10 Observations de l'historique - Transformateur de courant (suite et fin)

État	Observations d'apprentissage (valeurs réelles)
Réchauffement thermique	280,1500,150,1200,140 228,380,82,1012,19 50,90,18,260,5.9 48,230,160,810,7 90,160,54,330,29 168,1353,581,3281,63 577,3541,521,2928,7 1400,3000,560,3500,4 47,106,28.7,242.2,6.35 220,340,42,480,14 380,190,30,280,22 30,62,60,460,3.4 770,1420,401,1452,3 17000,110000,84000,89000,16000 81,130,74,230,2.9 130,440,180,730,0 70,69,29,241,10 130,440,180,730,0 3606.4,1182,328.4,1604.8,6.3 249,726,278,938,0 170,330,77,340,13 170,320,53,520,3.2 11,88,83,250,8.5 1000,4300,1100,5400,24 613,3240,1432,2788,0
État normal	10,4,3,33,6 8.5,7.2,4.3,3.9,3.5 32,31,7.5,50,1.1 13.5,1.7,1.2,0.6,0 9.87,2.49,0.79,4.06,4.8

Dans l'étude énoncée, les données historiques ont servi à choisir aléatoirement des données pour former des ensembles d'apprentissage et de tests.

50 observations historiques du transformateur principal de 500kV de la sub-station de Pingguo de la compagnie de génération électrique du Sud de la Chine, ont été utilisées pour l'apprentissage du LAD et ces données sont présentées dans le Tableau 4.10. Parmi cet échantillon de 50 observations, il existe 25 observations de réchauffement thermique, 15 observations de décharge de haute énergie, 5 observations de l'état normal et 5 observations de décharge de basse énergie.

La méthode LAD permet d'identifier des observations appartenant à un de deux états possibles. Bien qu'elle puisse être adaptée à traiter davantage d'états, ceci est en dehors des buts initiaux du projet. C'est la raison pour laquelle nous avons conduit 3 essais séparément, dont l'état négatif correspond toujours à l'état normal alors que l'état positif correspond à un de trois états de défaillance (décharge de haute énergie, décharge de basse énergie et réchauffement thermique).

Les fichiers en format binaire sont présentés à l'Annexe C, tant pour l'ensemble d'apprentissage que comme celui de tests. Voici une synthèse quantitative du processus de conversion binaire.

Essai 1 – États de haute énergie et normal : 39 indicateurs binaires.

Essai 2 – États de basse énergie et normal : 35 indicateurs binaires.

Essai 3 – États de réchauffement thermique et normal : 29 indicateurs binaires

Le Tableau 4.11 présente une synthèse de résultats de l'étape d'apprentissage.

Tableau 4.11 Synthèse de l'étape d'apprentissage - Transformateur de courant

États	Degré maximal de patrons permis dans l'algorithme LAD	# Patrons		Temps d'exécution
		Positifs	Négatifs	
Haute énergie et normal	3	397	216	18 minutes et 40 secondes
	4	468	408	35 minutes et 25 secondes
Basse énergie et normal	3	239	172	14 minutes et 32 secondes
	4	264	201	22 minutes et 6 secondes
Réchauffement thermique et normal	3	91	110	3 minutes et 10 secondes
	4	113	119	5 minutes et 50 secondes

Il faut noter que plus le degré d'un patron est petit, plus il est générique et plus il couvre davantage d'observations qu'un patron d'un degré d'ordre plus élevé. C'est ainsi que l'ajout de patrons de degré 4 n'apporte qu'un raffinement dans la classification de cas spécifiques, mais les résultats sont sensiblement les mêmes que ceux trouvés lorsqu'on se limitait aux patrons de degré 3. On a donc privilégié l'utilisation de patrons d'un degré maximal de 3.

L'état associé à chaque observation de l'ensemble de test était connu d'avance, mais le fichier qui a été utilisé dans les essais n'inclut pas l'état bien entendu. L'ensemble de tests consiste en 25 observations, dont 13 appartiennent à l'état de réchauffement thermique, deux à l'état décharge de haute énergie, quatre à l'état normal et six observations à l'état de décharge de basse énergie. Ces données sont présentées dans le Tableau 4.12.

Tableau 4.12 Observations de tests - Transformateur de courant

État	Observations de test
Décharge de haute énergie	293,50,13,115,120 443,85,9.5,103,174
Décharge de basse énergie	35,25,0,23,22 980,73,58,12,0 160,90,27,17,5 176,206,47.7,75.7,68.7 565,93,34,47,0 150,53,34,20,0
Réchauffement thermique	73,520,140,1200,6 15,125,29,574,7 2844,8517,4422,10196,39 8,631,254,2020,39 42,97,157,600,0 120,120,33,84,0.55 117,357,92,468,4 766,993,116,665,4 5,217,69,523,6 80,153,42,276,18 16,237,92,470,0 0,434,226,387,0 86,110,18,92,7.4

Tableau 4.12 Observations de tests - Transformateur de courant (suite et fin)

État	Observations de test
État normal	10,4,3,33,6 14.7,3.8,10.5,2.7,0.2 6.7,10,11,71,3.9 0.33,0.26,0.04,0.27,0

Nous avons soumis l'ensemble de tests à la fonction de classification, une synthèse de résultats obtenus est présentée dans le Tableau 4.13, le Tableau 4.14 et le Tableau 4.15.

Tableau 4.13 Résultats de classification du 1^{er} essai : État de décharge de haute énergie et normal - Transformateur de courant

#Obs.	Valeur de résultat de l'étape de classification	État connu à priori
1	0.668968	Positif
2	0.668968	Positif
3	-0.758184	Négatif
4	0.112772	Négatif
5	0.614056	Négatif
6	-0.473631	Négatif

Tableau 4.14 Résultats de classification du 2^{ème} essai : États de décharge de basse énergie et normal - Transformateur de courant

#Obs.	Valeur de résultat de l'étape de classification	État connu à priori
1	0.671545	Positif
2	0.627642	Positif
3	0.362221	Positif
4	0.425915	Positif

Tableau 4.14 Résultats de classification du 2^{ème} essai : États de décharge de basse énergie et normal - Transformateur de courant (suite et fin)

#Obs.	Valeur de résultat de l'étape de classification	État connu à priori
5	0.543089	Positif
6	0.627642	Positif
7	-0.695565	Négatif
8	-0.41525	Négatif
9	-0.465778	Négatif
10	-0.396197	Négatif

Tableau 4.15 Résultats de classification du 3^{ème} essai : États de réchauffement thermique et normal - Transformateur de courant

#Obs.	Valeur de résultat de l'étape de classification	État connu à priori
1	0.679663	Positif
2	0.268104	Positif
3	0.839483	Positif
4	-0.00417577	Positif
5	0.307196	Positif
6	0.26557	Positif
7	0.690959	Positif
8	0.690959	Positif
9	-0.332265	Positif
10	0.839483	Positif
11	-0.458796	Positif
12	-0.427404	Positif
13	0.797858	Positif

Tableau 4.15 Résultats de classification du 3^{ème} essai : États de réchauffement thermique et normal - Transformateur de courant (suite et fin)

#Obs.	Valeur de résultat de l'étape de classification	État connu à priori
14	-0.457143	Négatif
15	-0.583673	Négatif
16	-0.497184	Négatif
17	-0.552282	Négatif

Pour trouver les valeurs limites à partir desquelles une observation de test est classifiée comme négative ou positive, nous suivons la même procédure que dans l'application précédente, c'est-à-dire que nous soumettons l'ensemble d'apprentissage à la fonction de classification, pour arriver à ces valeurs seuil. Le Tableau 4.16 présente une synthèse de l'étape de classification et les valeurs seuils déduits de l'ensemble d'apprentissage (en gras et majuscule, les résultats corrects de classification).

Tableau 4.16 Résultats de classification - Transformateur de courant

# Essai	Décision de classification	# Obs.	Résultat	État déduit	État réel
Essai 1	Positif si ≥ 0.11 . Négatif si ≤ 0.01	1	0.668968	POSITIF	Positif
		2	0.668968	POSITIF	Positif
		3	-0.758184	NÉGATIF	Négatif
		4	0.112772	Positif	Négatif
		5	0.614056	Positif	Négatif
		6	-0.473631	NÉGATIF	Négatif

Tableau 4.16 Résultats de classification - Transformateur de courant (suite)

# Essai	Décision de classification	# Obs.	Résultat	État déduit	État réel
Essai 2	Positif si ≥ -0.14 Négatif si ≤ -0.24	1	0.671545	POSITIF	Positif
		2	0.627642	POSITIF	Positif
		3	0.362221	POSITIF	Positif
		4	0.275915	POSITIF	Positif
		5	0.543089	POSITIF	Positif
		6	0.627642	POSITIF	Positif
		7	-0.695565	NÉGATIF	Négatif
		8	-0.41525	NÉGATIF	Négatif
		9	-0.465778	NÉGATIF	Négatif
		10	-0.396197	NÉGATIF	Négatif

Tableau 4.16 Résultats de classification - Transformateur de courant (suite et fin)

# Essai	Décision de classification	# Obs.	Résultat	État déduit	État réel
Essai 3	Positif si ≥ -0.08 Négatif si ≤ -0.18	1	0.679663	POSITIF	Positif
		2	0.268104	POSITIF	Positif
		3	0.839483	POSITIF	Positif
		4	-0.0041757	POSITIF	Positif
		5	0.307196	POSITIF	Positif
		6	0.26557	POSITIF	Positif
		7	0.690959	POSITIF	Positif
		8	0.690959	POSITIF	Positif
		9	-0.332265	Négatif	Positif
		10	0.839483	POSITIF	Positif
		11	-0.458796	Négatif	Positif
		12	-0.427404	Négatif	Positif
		13	0.797858	POSITIF	Positif
		14	-0.457143	NÉGATIF	Négatif
		15	-0.583673	NÉGATIF	Négatif
		16	-0.497184	NÉGATIF	Négatif
		17	-0.552282	NÉGATIF	Négatif

Tableau 4.17 Synthèse de performance de classification - Transformateur de courant

Essai	# Obs.	% Fausses Alarmes	% Risque défaillance non détecté	% Erreur Total	% Efficacités de classification	% Indécision
1	6	33,33%	0%	33,33%	66,67%	0%
2	10	0%	0%	0%	100%	0%
3	17	0%	17,64%	17,64%	82,35%	0%

Nous remarquons que les résultats sont très révélateurs au sujet du risque de défaillance, car au cours de deux premiers essais, toutes les observations pour lesquelles il existe de défaillances ont été détectées. Seulement, au cours du troisième essai, le risque de défaillance non détecté est de 17,64%.

L'erreur liée aux fausses alarmes n'est présente qu'au cours du premier essai (33,33%) et dans le restant des cas, elle est nulle. La moyenne d'efficacité de classification de l'algorithme dans cette application est de 28 observations classifiées correctement sur un total de 33, soit de 84,8%.

Lors de travaux subséquents, il serait très important d'implémenter une fonctionnalité qui permette de passer de l'espace binaire des indicateurs présents dans les patrons générés vers les indicateurs originaux du domaine du problème. Ceci permettrait non seulement de comprendre le phénomène en question, mais aussi de comprendre plus en détails pourquoi certaines observations ne sont pas classifiées correctement.

Dans la dernière section de conclusions, nous allons décrire les principales idées de recherches futures en vue d'augmenter la précision du logiciel LAD-CBM, et d'améliorer le pouvoir explicatif des connaissances extraites. Nous aborderons également les principaux défis relevés dans la conception du logiciel LAD-CBM.

CONCLUSIONS ET RECHERCHES FUTURES

La maintenance conditionnelle préconise une approche proactive qui se sert de la surveillance des indicateurs de l'état d'une machine et l'analyse de ces données pour recommander des actions de maintenance et/ou réparation. La méthode LAD appliquée à la maintenance conditionnelle permet d'analyser un grand volume de données de l'historique d'une machine afin d'extraire des connaissances sous la forme de patrons sur les états où le risque de défaillance est élevé.

L'auteur a étudié l'utilisation du LAD comme technique d'extraction de connaissances et de classification d'observations dans le domaine de la maintenance conditionnelle et construit les fondements d'un logiciel qui automatise son application. Des essais ont été conclus sur deux ensembles d'observations de maintenance provenant de machines différentes et les résultats sont satisfaisants. Il est cependant possible d'améliorer la performance de l'algorithme en implémentant certains modules de support et optimisation.

Les résultats de ce travail de recherche sont de grande utilité dans un contexte industriel, notamment à cause du grand volume de données de maintenance générées à partir de capteurs dans les milieux industriels actuels et le besoin de faire un suivi de l'état des machines.

Au cours des expériences menées dans cette étude, nous avons expérimenté les avantages considérables d'avoir conçu le logiciel, car même avec un nombre élevé d'observations et d'indicateurs, l'extraction de connaissances se fait dans un petit intervalle de temps et la classification de nouvelles observations se fait en moins d'une seconde.

Ceci veut dire que même si les conditions intrinsèques des machines étudiées changent, ou si l'algorithme doit être adapté à de nouvelles machines, on peut extraire des connaissances sous forme de patrons de façon rapide et automatisée (en autant qu'on fournisse l'état pour les observations de l'historique) pour caractériser les machines au moment présent et la classification des nouvelles observations peut être effectuée de façon immédiate.

Il y a d'autres méthodes d'exploration de données mais l'utilisation du LAD à la maintenance conditionnelle présente l'avantage d'agir non seulement en tant qu'outil de prédiction de l'état de la machine mais permet aussi de faire l'extraction de connaissances sous un format compréhensible pour les humains.

Au niveau académique, cet avantage à lui seul est extrêmement intéressant à cause de conclusions issues de l'interprétation des connaissances extraites qui permettent de comprendre les phénomènes sous-jacents de défaillance.

À continuation, nous présentons une synthèse des principaux accomplissements dans la conception du logiciel LAD-CBM. Des améliorations et de recherches futures sont également proposées.

Au cours de ce projet, nous avons développé un outil informatique permettant d'analyser un fichier de données brutes dont on connaît l'état à priori (historique) pour ensuite déterminer l'état jusque-là inconnu à ce jour des nouvelles observations. La première étape consiste à faire une conversion binaire des observations de l'historique pour ensuite générer des expressions logiques (patrons) formées d'indicateurs binaires. Ces patrons décrivent la structure sous-jacente de données et constituent la base de la fonction de classification.

Pour trouver les fonctions pseudo-booléennes qui permettent de classifier des nouvelles observations à partir de patrons trouvés, on assigne un poids à chaque patron selon le nombre d'observations de l'ensemble d'apprentissage qu'il couvre. La fonction de classification serait une somme pondérée de l'évaluation des observations par rapport aux patrons. Le choix de valeurs seuils de classification pour l'état positif et négatif se fait de façon empirique de manière à réduire le risque de non-détection d'une anomalie tout en diminuant le nombre d'observations sur lesquelles le classificateur n'apporte pas de certitude (zone d'indécision).

Un des principaux défis dans la construction du logiciel LAD-CBM, était de choisir parmi le grand ensemble de modules et fonctionnalités abordés dans la théorie (Mayoral, E., 1998), ceux qui étaient absolument fondamentaux pour construire un logiciel fonctionnel de base. Bien que les modules aient été choisis avant à l'étape de conception, le module de conversion binaire n'a été implémenté qu'après avoir conçu et

même fait plusieurs essais du logiciel. Les avantages de l'automatisation de la conversion binaire dépassent largement l'effort de développement.

Dans un premier temps, il y a une grande économie de temps car la conversion binaire d'un ensemble d'observations comme le premier essai de l'exemple du transformateur de courant qui contient 5 indicateurs à l'origine mais qui résulte en 39 indicateurs binaires peut prendre plus de deux heures, si on le fait manuellement. Il y a aussi une possibilité d'erreurs de calcul considérable.

Lorsque la conversion binaire a été développée (effort de 4 heures-personne), elle permet de faire une conversion binaire équivalente de façon totalement automatisée en moins de 2 secondes sans risque d'erreur.

Un autre défi relevé est l'adaptation de concepts théoriques du LAD à la pratique. En effet, le sujet de l'analyse combinatoire et la croissance exponentielle du nombre de patrons générés avec le nombre d'indicateurs est une problématique intéressante. Lors des premiers essais, lorsqu'on essayait de générer des patrons de degré 4 au maximum, l'algorithme ne terminait pas l'extraction après 7 heures d'exécution. Nous avons alors conclu qu'il fallait s'adresser au sujet de l'explosion combinatoire.

Lors de la recherche d'alternatives de solutions, nous avons analysé toutes les étapes précédentes à la génération de patrons. Nous nous sommes rendus compte que la conversion binaire telle que proposée théoriquement (Hammer P.L., et al. 2000) génère un nombre trop élevé d'indicateurs binaires par rapport au nombre d'indicateurs de base. C'est ainsi qu'une technique réductrice du nombre d'indicateurs binaires a été proposée,

voir section 2.1.1 La conversion binaire de données. Cette réduction nous a permis de réduire considérablement le nombre d'indicateurs binaires et aussi de réduire le temps de génération de patrons d'au-delà de 7 heures, à un maximum de temps d'environ 19 minutes. Lors de la génération de patrons, le choix de structures de données appropriées à la navigation de candidats est essentiel.

La littérature existante ne décrit guère comment choisir les valeurs seuils à partir desquelles une observation est classifiée comme étant positive ou négative, mais elle affirme que ces valeurs seuils dépendent du problème spécifique traité. Nous avons mené une analyse de l'incidence de fausses alarmes (faux positifs) et des états de risque de défaillance non détectées, faux négatifs, afin de déterminer l'erreur acceptable dans le contexte de maintenance conditionnelle. Ensuite, avec ces objectifs, nous avons établi une zone d'indécision qui limitait les zones de classification positive et négative et qui tenait compte du fait qu'en maintenance, la non-détection d'une observation associée à l'état de risque de défaillance et plus sévère que le fait de générer une fausse alarme.

Nous voulons également noter que le logiciel n'a pas nécessité d'une synchronisation pour passer d'une application à l'autre et que les étapes suivies ont été identiques tant dans l'exemple du mécanisme de roulement que dans le cas du transformateur de courant. Cette caractéristique était un des buts du présent travail qui visait le développement d'un outil d'exploration dans le domaine de la maintenance conditionnelle, qui possède un bon taux de précision au niveau de la classification sans

que l'utilisateur ne soit obligé de faire de synchronisation de la méthode par rapport aux spécificités du problème traité.

L'outil informatique développé est un prototype de développement (voir Figure 3.1 Modules du logiciel LAD-CBM) qui devrait être adapté afin d'améliorer sa précision et le type de problèmes de maintenance conditionnelle qui peuvent être traités. Présentement, l'outil ne permet de classifier que des observations appartenant à un des deux états. Cependant, l'approche préconisée peut être adaptée à plus de deux états, ce qui serait utile non seulement pour détecter un risque de défaillance, mais également le type de défaillance.

La génération de patrons est une opération lente ; au-delà du degré 4 et pour un nombre d'indicateurs binaires supérieurs à 40 ainsi qu'un historique de 20 observations ou plus, la génération de patrons peut prendre au-delà de 20 minutes. L'obtention d'un ensemble de support (*covering set* en anglais) pour représenter les mesures des observations en format binaire serait donc l'amélioration de performance la plus importante qui pourrait être incorporée au logiciel LAD-CBM. Cet ensemble de support consiste à éliminer le plus grand nombre d'indicateurs binaires qui sont redondants, tout en assurant le maintien de la propriété fondamentale qui stipule qu'aucune observation ne peut pas être positive et négative en même temps. (Hammer P., et al., 2000)

En vue d'obtenir une couverture totale des données de l'historique, il serait également judicieux d'implémenter l'algorithme de génération descendante (du degré maximal jusqu'aux patrons de plus petit degré) de patrons pour les observations pour lesquelles

on n'a pas trouvé de patrons lorsque l'on a exécuté l'algorithme de génération de patrons ascendante jusqu'à un certain degré. Dans le cas du travail actuel, on a généré des patrons jusqu'au degré où toutes les observations étaient couvertes, tout en acceptant une longue durée d'exécution de l'étape d'apprentissage.

Il serait également intéressant d'étudier l'effet de l'exclusion systématique de patrons redondants (de patrons qui couvrent des observations que d'autres patrons couvrent déjà) sur la qualité de la fonction de classification.

Lorsque les observations sont des mesures nominales ou sont déjà en format binaire, l'interprétation de patrons est assez directe car les indicateurs contenus dans les patrons sont les mêmes ou sont faciles à déduire à partir de ceux du domaine du problème. Cependant, si les observations possèdent des indicateurs réels, l'interprétation des connaissances extraites sous forme de patrons n'est pas directe, car ces derniers contiennent des indicateurs binaires différents des indicateurs du problème. Il faudrait donc construire une fonction qui permettrait de faire le lien entre un ou plusieurs indicateurs binaires et les indicateurs originaux, voir section 2.1.1 . La conversion binaire de données. Ceci aiderait énormément à la compréhension du phénomène de défaillance ainsi que de résultats de classification.

L'outil informatique bénéficierait également d'avoir des outils graphiques afin de comprendre la couverture de patrons et les résultats de la classification des observations de tests. Lorsque le nombre de données devient considérable, une analyse graphique afin de comprendre où se situe une observation de test par rapport aux observations

d'apprentissage peut aider à prendre de meilleures décisions de classification, voir la Figure 2.2.

RÉFÉRENCES

- ALLGOOD, G.O., UPADHYAYA, B.R., (2000), *A model-based high frequency matched filter arcing diagnostic system based on principal component analysis (PCA) clustering*, Applications and Science of Computational Intelligence III, vol. 4055, Bellingham, pp. 430-440.
- AL-NAJJAR, B., ALSYOUF, I., (2004), *Enhancing a company's profitability and competitiveness using integrated vibration-based maintenance: A case study*. Journal of operational research, Vol. 15, pp. 643-657.
- AMJADY, N., EHSAN, M., (1999), *Evaluation of power systems reliability by an artificial neural network*. IEEE Transactions on Power Systems. Vol 14, No. 1.
- BAIG, M.F., SAYEED, N, (1998), *Model-based reasoning for fault diagnosis of twin-spool turbofans*, Proceedings of the Institution of Mechanical Engineers, Part G. Journal of Aerospace Engineering, 212, pp. 109-116.
- CENDROWSKA, J., (1987), *PRISM: An algorithm for inducing modular rules*, International Journal of Man-Machine Studies, 1987, 27, pp. 349-370.
- DEKKER, R., (1996), *Applications of maintenance optimization models: a review and analysis*, Reliability Engineering and System Safety, Vol. 51, pp. 229-240.

FUGATE, M.L., SOHN, H., FARRAR, C.R., (2001), *Vibration-based damage detection using statistical process control*, Mechanical Systems and Signal Processing, Vol. 15, pp. 707-721.

GANYUN, L., HAOZHONG C., HAIBAO Z., Lixin D., (2005), *Fault diagnosis of power transformer based on multi-layer SVM classifier*, Electric Power Systems Research, Vol. 75, pp. 9-15.

GERTLER, J.J., (1998), Fault Detection and Diagnosis in Engineering Systems, Marcel Dekker, New York.

HOLMBERG, K., FOLKESON, A., (1991), *Operational Reliability and Systematic Maintenance*. CRC Press, pp. 178.

HAMMER, P.L., BOROS, E., IBARAKI, T., KOGAN, A., MAYORAZ, E., MUCHNIK, I., (2000), *An implementation of Logical Analysis of Data*, IEEE Transactions on Knowledge and Data Engineering, Vol. 12, No. 2.

HAMMER, P.L., CRAMA, Y., IBARAKI, T., (1988), *Cause-Effect Relationships and Partially Defined boolean functions*, Annals of Operations Research, Vol. 16, pp. 299-325.

HONG, W.C., PAI, P.F., (2006), *Predicting engine reliability by support vector machines*, International journal of advanced manufacturing technology. Vol.28, pp.154-161.

JARDINE, A. K. S., LIN, D., BANJEVIC, D., (2006), *A review on machinery diagnostics and prognostics implementing condition-based maintenance*, Mechanical Systems and Signal Processing, vol. 20, pp. 1483-1510.

KIRIANAKI, N.V., YURISH, S.Y., SHPAK, N.O., DEYNEGA, V.P., 2002, *Data Acquisition and Signal Processing for Smart Sensors*, Wiley, Chichester, West Sussex, England.

LEE, J., ABUJAMRA, R., JARDINE, A.K.S., LIN D., BAJEVIC D., (2004), *An integrated platform for diagnostics, prognostics and maintenance optimization*, in: The IMS '2004 International Conference on Advances in Maintenance and in Modeling, Simulation and Intelligent Monitoring of Degradations, Arles, France.

LEE, S.J., SIAU, K., (2001), *A review of data mining techniques*. Industrial Management & Data Systems. MCB University Press. Vol. 1001, pp 41-46.

LI, Z., WU, Z., XU, Y., (2005), *Hidden Markov model based fault diagnostics method in speed-up and speed-down process for rotating machinery*, Mechanical Systems and Signal Processing , Vol. 19, pp. 329-339.

LIU , Q., WANG, H.-P., (2001), *A case study on multisensor data fusion for imbalance diagnosis of rotating machinery* , (AI EDAM) Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol. 15 ,pp. 203-210.

MARTIN, K.F., (1994), *A review by discussion of condition monitoring and fault-diagnosis in machine-tools*, International Journal of Machine Tools and Manufacture, Vol. 34, pp. 527-551.

MAYORAZ, E., (1998), *C++ Tools for Logical Analysis of Data*, Rutgers University's Center for Operations Research, New Jersey.

MCKONE, K., WEISS, E., (1998), *TPM: Planned and autonomous maintenance: Bridging the gap between practice and research*, Production and Operations Management, Vol. 4, pp. 335-351.

PINTELON, L.M., GELDERS, L.F., (1992), *Maintenance management decision making*, European Journal of Operations Research, Vol. 58, pp. 301-317.

PREGIBON D., (1996), *Data mining*, Statistical Computing and Graphics. Vol. 7, pp.8.

RAHEJA, D., LLINAS, J., NAGI, R., ROMANOWSKI, C., (2006), *Data fusion/data mining-based architecture for condition-based maintenance*, International Journal of Production Research, Vol. 44, No. 14, pp. 2869-2887.

SAMANTA, B., AL-BALUSHI, K.R., (2003), *Artificial neural network based fault diagnostics or rolling element bearings using time-domain features*, Mechanical Systems and Signal Processing, Vol. 17, pp. 317-328.

SEKHAR, A.S., (2004), *Model-based identification of two cracks in a rotor system*, Mechanical Systems and Signal Processing, Vol. 18, pp. 977-983.

SIDDIQUE, A., YADAVA, G.S., SINGH, B., (2003), *Applications of artificial intelligence techniques for induction machine stator fault diagnostics: Review*, Proceedings of the IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives. New York, pp. 29-34.

SIMANI, S., FANTUZZI, C., PATTON, R.J., (2003), *Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques*, Springer, London, 2003.

SOHN, H., WORDEN, K. , FARRAR, C.R., (2002), *Statistical damage classification under changing environmental and operational conditions*, Journal of Intelligent Material Systems and Structures, Vol. 13, No. 9, pp 561-574

SPOERRE J.K., (1997), *Application of the cascade correlation algorithm (CCA) to bearing fault classification problems*. Computers in Industry, Vol. 32, pp. 295-304.

STELLMAN, C.M., EWING, K.J., BUCHOLTZ, F., AGGARWAL, I.D., (1999), *Monitoring the degradation of a synthetic lubricant oil using infrared absorption, fluorescence emission and multivariate analysis: A feasibility study*, Lubrication Engineering, Vol. 55, pp 42-52.

SUN, Q., CHEN, P., ZHANG, D., XI, F., (2004), *Pattern recognition for automatic machinery fault diagnosis*, Journal of Vibration and Acoustics, Vol. 124.

TANDON, N., CHOUDRY, A., (1999), *A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings*, Tribology International, Vol. 32, issue 8, pp. 469-480.

YOON, Y.H., YOON, E.S., CHANG, K.S., (1993), *Process fault diagnostics using the integrated graph model*, On-Line Fault Detection and Supervision in the Chemical Process Industries, Oxford, pp. 89-94.

WANG, W., (2000), *Model to determine the optimal critical level and the monitoring intervals in condition based maintenance*, International Journal of Production Research, Vol. 38, No. 6, pp 1425-1436.

WANG, W., (2007), *A two-stage prognosis model in condition based maintenance*, European Journal of Operational Research, Vol. 182, pp. 1177-1187.

WEN, Z.Y., Crossman, J., CARDILLO, J., MURPHEY, Y.L., (2003), *Case based reasoning in vehicle fault diagnostics*, Proceedings of International Joint Conference on Neural Networks, Vols. 1-4, New York, pp. 2679-2684.

WILLIAMS, H., ZOBEL, J., 1999, *Compressing Integers for Fast File Access*, The Computer Journal, Vol. 42, No.3.

WIREMAN, T., 1990, *World class maintenance management*, Industrial Press, New York.

ANNEXE A

ÉTUDE DE PRESCRIPTION DE VERRES DE CONTACT

Tableau A.1 Observations en format binaire - Étude de verres de contact

# Obs.	État	Âge		Type de verre	Astigmatique	Production de larmes
		b_0	b_1	b_2	b_3	b_4
1	1	0	0	0	0	0
2	0	0	0	0	0	1
3	1	0	0	0	1	0
4	0	0	0	0	1	1
5	1	0	0	1	0	0
6	1	0	0	1	0	1
7	1	0	0	1	1	0
8	0	0	0	1	1	1
9	1	0	1	0	0	0
10	0	0	1	0	0	1
11	1	0	1	0	1	0
12	0	0	1	0	1	1
13	1	0	1	1	0	0
14	0	0	1	1	0	1
15	1	0	1	1	1	0
16	1	0	1	1	1	1
17	1	1	1	0	0	0
18	1	1	1	0	0	1
19	1	1	1	0	1	0
20	0	1	1	0	1	1
21	0	1	1	1	0	1
22	0	1	1	1	0	1
23	1	1	1	1	1	0
24	1	1	1	1	1	1

ANNEXE B

APPLICATION DE CBM AU MÉCANISME DE ROULEMENT

Résultats de l'étape d'apprentissage (génération de patrons)

Tableau B.1 Patrons générés au cours du premier essai - Mécanisme de roulement

Essai 1 - Génération de patrons 49 patrons négatifs, 28 patrons positifs.
Negative pattern #1: NOT 1 Observations covered by this pattern: 4,5,6
Negative pattern #2: NOT 2 Observations covered by this pattern: 4,5,6
Negative pattern #3: 4 Observations covered by this pattern: 4
Negative pattern #4: 7 Observations covered by this pattern: 5,6
Negative pattern #5: 9 Observations covered by this pattern: 4,5,6
Negative pattern #6: 12 Observations covered by this pattern: 4
Negative pattern #7: 15 Observations covered by this pattern: 5,6
Negative pattern #8: 17 Observations covered by this pattern: 4
Negative pattern #9: 20 Observations covered by this pattern: 5,6
Negative pattern #10: 3,NOT 5 Observations covered by this pattern: 4
Negative pattern #11: 3,NOT 6 Observations covered by this pattern: 4,5,6
Negative pattern #12: 3,NOT 10 Observations covered by this pattern: 4,5,6
Negative pattern #13: 3,NOT 13 Observations covered by this pattern: 4
Negative pattern #14: 3,NOT 14 Observations covered by this pattern: 4,5,6
Negative pattern #15: 3,NOT 18 Observations covered by this pattern: 4

Tableau B.1 Patrons générés au cours du premier essai - Mécanisme de roulement (suite)

Essai 1 - Génération de patrons 49 patrons négatifs, 28 patrons positifs.
Negative pattern #16: 3,NOT 19 Observations covered by this pattern: 4,5,6
Negative pattern #17: 5,NOT 6 Observations covered by this pattern: 5,6
Negative pattern #18: 5,NOT 10 Observations covered by this pattern: 5,6
Negative pattern #19: 5,NOT 14 Observations covered by this pattern: 5,6
Negative pattern #20: 5,NOT 19 Observations covered by this pattern: 5,6
Negative pattern #21: NOT 5,8 Observations covered by this pattern: 4
Negative pattern #22: NOT 5,11 Observations covered by this pattern: 4
Negative pattern #23: NOT 5,16 Observations covered by this pattern: 4
Negative pattern #24: NOT 6,8 Observations covered by this pattern: 4,5,6
Negative pattern #25: NOT 6,11 Observations covered by this pattern: 4,5,6
Negative pattern #26: NOT 6,13 Observations covered by this pattern: 5,6
Negative pattern #27: NOT 6,16 Observations covered by this pattern: 4,5,6
Negative pattern #28: NOT 6,18 Observations covered by this pattern: 5,6
Negative pattern #29: 8,NOT 10 Observations covered by this pattern: 4,5,6
Negative pattern #30: 8,NOT 13 Observations covered by this pattern: 4
Negative pattern #31: 8,NOT 14 Observations covered by this pattern: 4,5,6
Negative pattern #32: 8,NOT 18 Observations covered by this pattern: 4
Negative pattern #33: 8,NOT 19 Observations covered by this pattern: 4,5,6
Negative pattern #34: NOT 10,11 Observations covered by this pattern: 4,5,6
Negative pattern #35: NOT 10,13 Observations covered by this pattern: 5,6

Tableau B.1 Patrons générés au cours du premier essai - Mécanisme de roulement (suite)

Essai 1 - Génération de patrons	
49 patrons négatifs, 28 patrons positifs.	
Negative pattern #36:	NOT 10,16 Observations covered by this pattern: 4,5,6
Negative pattern #37:	NOT 10,18 Observations covered by this pattern: 5,6
Negative pattern #38:	11,NOT 13 Observations covered by this pattern: 4
Negative pattern #39:	11,NOT 14 Observations covered by this pattern: 4,5,6
Negative pattern #40:	11,NOT 18 Observations covered by this pattern: 4
Negative pattern #41:	11,NOT 19 Observations covered by this pattern: 4,5,6
Negative pattern #42:	13,NOT 14 Observations covered by this pattern: 5,6
Negative pattern #43:	13,NOT 19 Observations covered by this pattern: 5,6
Negative pattern #44:	NOT 13,16 Observations covered by this pattern: 4
Negative pattern #45:	NOT 14,16 Observations covered by this pattern: 4,5,6
Negative pattern #46:	NOT 14,18 Observations covered by this pattern: 5,6
Negative pattern #47:	16,NOT 18 Observations covered by this pattern: 4
Negative pattern #48:	16,NOT 19 Observations covered by this pattern: 4,5,6
Negative pattern #49:	18,NOT 19 Observations covered by this pattern: 5,6
Positive pattern #1:	1 Observations covered by this pattern: 1,2,3
Positive pattern #2:	2 Observations covered by this pattern: 1,2,3
Positive pattern #3:	NOT 3 Observations covered by this pattern: 1,2
Positive pattern #4:	6 Observations covered by this pattern: 3
Positive pattern #5:	NOT 8 Observations covered by this pattern: 1,2
Positive pattern #6:	NOT 9 Observations covered by this pattern: 1,2,3
Positive pattern #7:	10 Observations covered by this pattern: 3

Tableau B.1 Patrons générés au cours du premier essai - Mécanisme de roulement (suite)

<p align="center">Essai 1 - Génération de patrons</p> <p align="center">49 patrons négatifs, 28 patrons positifs.</p>
Positive pattern #8: NOT 11 Observations covered by this pattern: 1,2
Positive pattern #9: 14 Observations covered by this pattern: 3
Positive pattern #10: NOT 16 Observations covered by this pattern: 1,2
Positive pattern #11: 19 Observations covered by this pattern: 3
Positive pattern #12: NOT 4,NOT 5 Observations covered by this pattern: 1,2
Positive pattern #13: NOT 4,NOT 7 Observations covered by this pattern: 1,2,3
Positive pattern #14: NOT 4,NOT 13 Observations covered by this pattern: 1,2
Positive pattern #15: NOT 4,NOT 15 Observations covered by this pattern: 1,2,3
Positive pattern #16: NOT 4,NOT 18 Observations covered by this pattern: 1,2
Positive pattern #17: NOT 4,NOT 20 Observations covered by this pattern: 1,2,3
Positive pattern #18: 5,NOT 7 Observations covered by this pattern: 3
Positive pattern #19: 5,NOT 15 Observations covered by this pattern: 3
Positive pattern #20: 5,NOT 20 Observations covered by this pattern: 3
Positive pattern #21: NOT 5,NOT 12 Observations covered by this pattern: 1,2
Positive pattern #22: NOT 5,NOT 17 Observations covered by this pattern: 1,2
Positive pattern #23: NOT 7,NOT 12 Observations covered by this pattern: 1,2,3
Positive pattern #24: NOT 7,13 Observations covered by this pattern: 3
Positive pattern #25: NOT 7,NOT 17 Observations covered by this pattern: 1,2,3
Positive pattern #26: NOT 7,18 Observations covered by this pattern: 3

Tableau B.1 Patrons générés au cours du premier essai - Mécanisme de roulement (suite et fin)

Essai 1 - Génération de patrons 49 patrons négatifs, 28 patrons positifs.	
Positive pattern #27:	NOT 12,NOT 13 Observations covered by this pattern: 1,2
Positive pattern #28:	NOT 12,NOT 15 Observations covered by this pattern: 1,2,3
Positive pattern #29:	NOT 12,NOT 18 Observations covered by this pattern: 1,2
Positive pattern #30:	NOT 12,NOT 20 Observations covered by this pattern: 1,2,3
Positive pattern #31:	13,NOT 15 Observations covered by this pattern: 3
Positive pattern #32:	13,NOT 20 Observations covered by this pattern: 3
Positive pattern #33:	NOT 13,NOT 17 Observations covered by this pattern: 1,2
Positive pattern #34:	NOT 15,NOT 17 Observations covered by this pattern: 1,2,3
Positive pattern #35:	NOT 15,18 Observations covered by this pattern: 3
Positive pattern #36:	NOT 17,NOT 18 Observations covered by this pattern: 1,2
Positive pattern #37:	NOT 17,NOT 20 Observations covered by this pattern: 1,2,3
Positive pattern #38:	18,NOT 20 Observations covered by this pattern: 3

ANNEXE C

APPLICATION DE CBM AU TRANSFORMATEUR DE COURANT

**Tableau C.1 Données d'apprentissage. État de décharge de haute énergie et état normal
- Transformateur de courant**

Données d'apprentissage - État de décharge de haute énergie et état normal
1,1,0,1,0,1,1,1,0,0,0,0,0,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1
1,1,1,0,0,0,1,0,1,0,1,0,1,0,1,1,0,0,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,1,0,0,0,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,1,0,0,0,1
1,1,1,0,0,0,1,0,1,0,1,1,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,1,0,0,0,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,1,0,0,0,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,1,0,0,0,1,0,1,1,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1,0,0,0,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,1,0,0,0,1,0,1,0,1,0,1,1,0,0,0,1,0,1,0,1,1,0,0,0,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,1
1,1,0,1,0,1,1,0,1,0,1,1,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,1,0,0,0,1
0,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,1,0,0

Tableau C.1 Données d'apprentissage. État de décharge de haute énergie et état normal - Transformateur de courant (suite et fin)

Données d'apprentissage - État de décharge de haute énergie et état normal
0,0,0,0,0,0,1,0,1,0,1,0,1,1,0,0,0,0,0,1,0,1,0,1,1,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0
0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1,0,0
0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
0,1,1,0,0,0,0,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0

Tableau C.2 Données de test. État de décharge de haute énergie et état normal
~ Transformateur de courant

[illegible]

**Tableau C.3 Données d'apprentissage - État de décharge de basse énergie et état normal
- Transformateur de courant**

Données d'apprentissage - État de décharge de basse énergie et état normal
1,1,0,1,1,0,1,0,1,0,1,1,0,1,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0
1,0,0,0,1,1,0,0,0,0,0,0,0,1,0,1,1,0,0,0,1,1,0,0,0,0,0,1,0,1,0,1,0,1
1,1,0,1,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1,0,1,0,1,1,0,0,0,0,0,0,0,0
1,1,0,1,1,0,1,0,1,0,1,0,1,1,0,1,0,1,1,1,0,0,0,0,0,1,0,1,0,1,1,0,0,0
1,1,0,1,1,0,1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,1,0,1,0,1,0,1
0,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,1,0,1,1,0,0,0,1,0,1,0,1,1,0
0,1,1,0,1,0,1,1,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,0,0,0,0
0,1,1,0,1,0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1,0,1,0,1,1,1,0,0,0,0,0,0,0
0,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
0,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,0,0,0,0

**Tableau C.5 Données d'apprentissage - État de réchauffement thermique et état normal
- Transformateur de courant (suite et fin)**

Données d'apprentissage - État de réchauffement thermique et état normal
1,1,0,1,0,1,0,1,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,1,0
0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,1,0,0,0,0,0,0
0,1,0,1,0,1,0,1,1,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0
0,1,0,1,1,0
0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,1,0,0,0,0

